

Análisis Cinemático y Dinámico de Robots Tipo Delta

David Zenteno Lara*

zDynamics | The Future is ROBOTICS

Índice

1. Introducción	4
1.1. Sobre la Notación Utilizada	4
1.2. ¿Cómo Citar este Documento?	5
2. Cinemática Directa	6
2.1. Análisis de la i -ésima cadena cinemática	7
2.2. Relación Geométrica	8
2.3. Cinemática Virtual	10
2.4. Solución para un Sistema con Tres Articulaciones	11
2.5. Algoritmo Numérico	15
2.6. Resultado Numérico de la Cinemática Directa	15
3. Cinemática Inversa	16
3.1. Matriz Jacobiana Geométrica	17
3.2. Forma Numérica de la Solución	21
3.3. Resultados	21
4. Cinemática Diferencial	22
4.1. Derivada de la Matriz Jacobiana Geométrica	23
4.2. Velocidad Instantánea de las Articulaciones	25
4.3. Velocidad Instantánea del Efector Final	26
4.4. Aceleración Instantánea de las Articulaciones	27
4.5. Aceleración Instantánea del Efector Final	28
5. Análisis Dinámico	29
5.1. Posiciones y Velocidades en el Robot	29
5.2. Energía Cinética y Potencial	34
5.3. Matrices de Rotación y Tensores de Inercia de los Cuerpos Rígidos	36
5.3.1. Cuerpos Rígidos l_i	37
5.4. Modelo Matemático	38
5.4.1. Restricciones Cinemáticas	39
5.4.2. Modelo en el Espacio de las Articulaciones	41
5.4.3. Modelo Auxiliar en el Espacio Cartesiano	46
5.5. Simulación Numérica	48

*Revisado por José Luis Boyzo Ríos

6. Sistema de Control	51
6.1. Control de Posición y Velocidad de las Articulaciones	51
6.2. Control de Posición y Velocidad del Efector Final	54
Apéndices	58
A.1. Cinemática Directa	58
A.2. Matriz Jacobiana Geométrica	59
A.3. Cinemática Inversa	60
A.4. Derivada de una Matriz Jacobiana Geométrica	61
A.5. Velocidad Instantánea de las Articulaciones	62
A.6. Velocidad Instantánea del Efector Final	63
A.7. Aceleración Instantánea de las Articulaciones	64
A.8. Aceleración Instantánea del Efector Final	65
A.9. Matrices Jacobianas Geométricas del Cuerpo Rígido l_i	66
A.10. Matriz de Rotación del Cuerpo Rígido l_i	67
A.11. Tensor de Inercia respecto al Marco Inercial	68
A.12. Matriz de Inercia del Robot	69
A.13. Matriz de Efectos Centrifugos y de Coriolis	70
A.14. Matriz de Efectos Gravitacionales	71
A.15. Torque de Restricción τ_r	72
A.16. Coeficientes $\bar{\lambda}$	73
A.17. Bloques del Modelo Dinámico en Simulink	74
A.18. Bloques del Sistema de Control de las Articulaciones en Simulink	76
A.19. Bloques del Sistema de Control Cartesiano en Simulink	77

Índice de figuras

1.1. Ejemplo de un robot paralelo tipo Delta [2]	4
1.2. Ejemplo de la notación utilizada.	4
2.1. Representación cinemática de un robot paralelo [2]	6
2.2. Ejemplo de separación angular entre las articulaciones de un robot paralelo	7
2.3. Centro Instantáneo de Rotación entre la i -ésima cadena cinemática y el efector final	9
2.4. Representación de los movimientos de la cadena $\{a_i, l_i\}$ con el efector final p_i a través del eslabón v_i , acotados en una esfera	10
2.5. Representación del acotamiento en los movimientos de los elementos que conforman un robot paralelo con tres articulaciones.	11
2.6. Simplificación de las esferas en planos que se intersecan.	13
2.7. Representación gráfica de la Cinemática Directa.	16
3.1. Vectores de posición de las uniones entre cuerpos rígidos en un Robot Delta	17
3.2. Representación gráfica de la Cinemática Inversa.	22
4.1. Velocidad Instantánea las Articulaciones	26
4.2. Velocidad Instantánea del efector final	27
4.3. Aceleración Instantánea de las Articulaciones	28
4.4. Aceleración Instantánea del Efector Final	29
5.1. Centro de Masa del Efector Final.	30
5.2. Punto de Unión entre la Base del Robot y el Cuerpo Rígido l_i	30
5.3. Punto Extremo del Cuerpo Rígido l_i	31
5.4. Punto Extremo del Cuerpo Rígido v_i	33
5.5. Ejemplo de un Cuerpo Rígido alineado con el Marco de Referencia Inercial.	36
5.6. Ejemplo de Transformación entre Marcos de Referencia.	36
5.7. Ejemplo de la Simplificación en la Geometría del Eslabón l_i .	38
5.8. Diagrama de Bloques del Modelo Dinámico.	49
5.9. Cálculo de $\bar{\lambda}$ y τ_r en Simulink.	49
5.10. Comportamiento de las Articulaciones y el Efector Final en Simulink	50
5.11. Coeficientes $\bar{\lambda}$ y torque de restricción τ_r .	51
6.1. Diagrama de Bloques del Sistema de Control.	53
6.2. Comportamiento Controlado de las Articulaciones y el Efector Final	54
6.3. Diagrama de Bloques del Sistema de Control Cartesiano.	56
6.4. Comportamiento Controlado de las Articulaciones y el Efector Final	57

1. Introducción

Los robots paralelos son cadenas cinemáticas cerradas cuyos movimientos son generados a través de articulaciones que se relacionan entre sí por medio de eslabones conectados en un punto común, conocido como efector final [1].

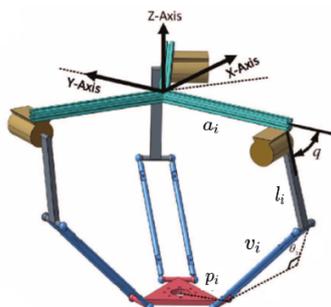


Figura 1.1: Ejemplo de un robot paralelo tipo Delta [2]

A diferencia de los robots que son cadenas cinemáticas abiertas (e.g. robots industriales), los grados de libertad de un robot paralelo no se definen solo por el número de articulaciones, sino por los movimientos que se pueden realizar en el efector final. Ya que este se mueve linealmente sobre los ejes (x, y, z) , se considera que un robot paralelo tiene tres grados de libertad, aunque pueden existir algunos sistemas que generen rotaciones en el efector final, lo que incrementa el número de grados de libertad del sistema.

1.1. Sobre la Notación Utilizada

El estudio de sistemas dinámicos con múltiples cuerpos rígidos requiere de una utilizar una notación que permita describir donde comienza y termina un vector, además de considerar desde qué punto es visto dentro de un espacio tridimensional. Para simplificar esto, se propone utilizar la siguiente notación

$$\vec{r}_{y/x}^z \rightarrow \vec{r}_{\text{hasta/desde}}^{\text{visto desde}}$$

Por ejemplo, si existe un cuerpo rígido que tiene un marco de referencia móvil i acoplado a él (figura 1.2), la posición de este último se describe como

$$\vec{r}_{i/0}^0 \rightarrow \vec{r}_{\text{termina/comienza}}^{\text{se ve desde}}$$

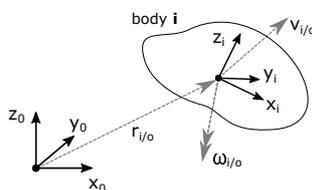


Figura 1.2: Ejemplo de la notación utilizada.

Esta notación se explica detalladamente en [3].

1.2. ¿Cómo Citar este Documento?

Al ser un documento abierto para cualquier lector (tipo *white paper*), este puede citarse de la siguiente manera

- Formato **APA**

Zenteno-Lara, D. (2022). *Análisis Cinemático y Dinámico de Robots Tipo Delta* [White paper]. México. [zDynamics.org](https://www.zdynamics.org)

- *Cita entre Paréntesis*: (Zenteno-Lara, 2022)
- *Cita Narrativa*: Zenteno-Lara (2022)

- Formato **IEEE**

D. Zenteno-Lara, "Análisis Cinemático y Dinámico de Robots Tipo Delta," México, [zDynamics.org](https://www.zdynamics.org), White Paper, 2022. [En línea]. Disponible en: <https://www.zdynamics.org/docs/ParalelosDelta.pdf>

- Formato **L^AT_EX**

```
@incollection{paraleloZ2022,  
  author      = "David Zenteno-Lara",  
  title       = "Análisis Cinemático y Dinámico de Robots Tipo Delta",  
  booktitle   = "Robótica: de la Cinemática al Control",  
  year        = 2022,  
  organization= "zDynamics.org",  
  address     = "México",  
  note        = "{Udemy}",  
  url         = "https://www.zdynamics.org/docs/ParalelosDelta.pdf",  
}
```

2. Cinemática Directa

Un robot paralelo está conformado por un conjunto de cadenas cinemáticas $\{a_i, l_i\}$ donde todas son iguales entre sí. Estas parten del marco de referencia inercial y terminan en el punto marcado en la siguiente figura

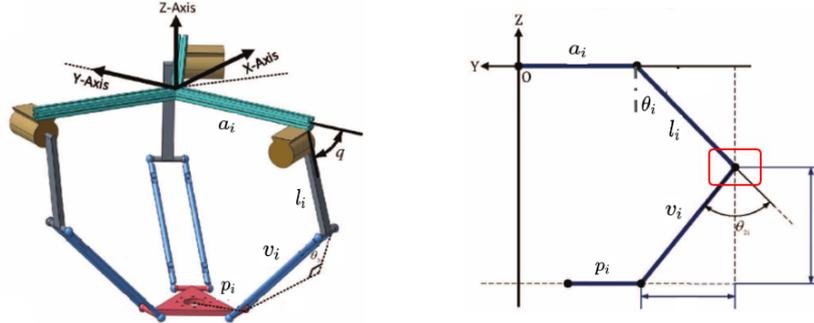


Figura 2.1: Representación cinemática de un robot paralelo [2]

Los eslabones que se encuentran debajo de la marca en la figura anterior no son considerados parte de la cadena cinemática, pero esto no significa que no sean afectados por los movimientos de las articulaciones. Estos eslabones, con una longitud fija $v_i \in \mathbb{R}$, se consideran *virtuales* porque el cálculo de sus movimientos depende del cálculo de las raíces de una ecuación, cuyos resultados pueden ser reales (positivos y negativos) e imaginarios. En general, el procedimiento para conocer la posición del efector final puede describirse, de forma generalizada, como se enlista a continuación:

1. Estudiar la cinemática directa $H_{i/0}^0$ de la i -ésima cadena cinemática, dependiente de la articulación alineada con cualquier eje del marco inercial
 - a) Rotar la matriz de transformación homogénea obtenida anteriormente de acuerdo al número de articulaciones en el robot
2. Definir la relación geométrica entre el extremo de la i -ésima cadena cinemática y el extremo del robot
3. Resolver las ecuaciones que definen la relación geométrica previamente establecida. El resultado será la posición del extremo del robot.

Sin embargo, antes de comenzar con el cálculo, es importante definir el número de articulaciones en el sistema y la separación angular entre ellas; comúnmente, estos robots constan de tres articulaciones, separadas $\frac{2}{3}\pi$ radianes como se muestra a continuación

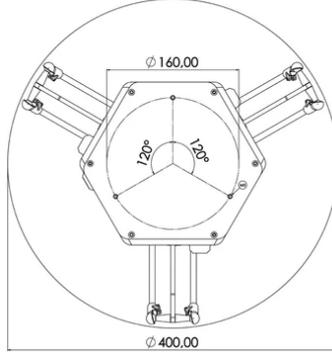


Figura 2.2: Ejemplo de separación angular entre las articulaciones de un robot paralelo

Se debe enfatizar que este análisis es válido para CUALQUIER ROBOT, sin embargo, los resultados SIMBÓLICOS, e.g. ecuación (2.8), son válidos ÚNICAMENTE para robots paralelos cuyo marco inercial tiene al eje z apuntando hacia arriba, y al eje y alineado con una de las tres articulaciones. De otra forma, debes seguir el mismo procedimiento, pero cambiando algunos términos en las ecuaciones (2.5) y (2.9) y en todas aquellas donde se involucre un movimiento respecto al marco inercial, lo que llevará a resultados completamente distintos a los mostrados aquí.

2.1. Análisis de la i -ésima cadena cinemática

Existen múltiples formas de analizar las cadenas cinemáticas $\{a_i, l_i\}$, ya sea de forma individual o en conjunto. La más sencilla es estudiando aquella cadena que se encuentra alineada con algún eje del marco inercial y replicar el resultado al multiplicarlo por una matriz de rotación. De forma generalizada, el número de cadenas cinemáticas se define como $i = 1, 2, 3, \dots$, además, basándose en la figura 2.1, se escoge la articulación alineada con el eje inercial y como el punto de partida del problema a resolver. Por lo tanto, la cinemática directa se puede definir con la siguiente ecuación

$$H_{i/0}^0 = T_{-a_i, y} R_{\theta_i, -x} T_{-l_i, z}, \quad (2.1)$$

donde $H_{i/0}^0$, $T_{-a_i, z}$, $R_{\theta_i, -x}$, $T_{-l_i, z} \in \mathbb{R}^{4 \times 4}$ son matrices de transformación homogénea, comenzando por la representación de la posición de la i -ésima cadena cinemática, una traslación $a_i \in \mathbb{R}$ sobre el eje y inercial, además de una rotación sobre el eje x inercial (en sentido negativo) para la articulación $\theta_i \in \mathbb{R}$ y una traslación sobre el eje z inercial en función de la longitud $l_i \in \mathbb{R}$ del eslabón (respectivamente).

$$T_{-a_i, y} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & -a_i \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (2.2)$$

$$R_{\theta_i, -x} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\theta_i) & \sin(\theta_i) & 0 \\ 0 & -\sin(\theta_i) & \cos(\theta_i) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (2.3)$$

$$\mathbf{T}_{-l_i, z} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -l_i \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (2.4)$$

Para generalizarlo al resto de las cadenas cinemáticas, basta con multiplicar el lado derecho de la ecuación por una matriz de rotación $\mathbf{R}_{\phi_i, z} \in \mathbb{R}^{4 \times 4}$ sobre el eje z :

$$\mathbf{H}_{i/0}^0 = \mathbf{R}_{\phi_i, z} \mathbf{T}_{-a_i, y} \mathbf{R}_{\theta_i, -x} \mathbf{T}_{-l_i, z}, \quad (2.5)$$

con $\phi_i \in \mathbb{R}$, que representa la separación angular de la i -ésima cadena cinemática respecto a la anterior, esto se puede interpretar como

$$\mathbf{R}_{\phi_i, z} = \begin{bmatrix} \cos(\phi_i) & -\sin(\phi_i) & 0 & 0 \\ \sin(\phi_i) & \cos(\phi_i) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (2.6)$$

entonces, la ecuación (2.5) se expande de la siguiente manera

$$\begin{aligned} \mathbf{H}_{i/0}^0 &= \mathbf{R}_{\phi_i, z} \mathbf{T}_{-a_i, y} \mathbf{R}_{\theta_i, x} \mathbf{T}_{-l_i, z} \\ &= \begin{bmatrix} \cos(\phi_i) & -\sin(\phi_i) & 0 & 0 \\ \sin(\phi_i) & \cos(\phi_i) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & -a_i \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\theta_i) & \sin(\theta_i) & 0 \\ 0 & -\sin(\theta_i) & \cos(\theta_i) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -l_i \\ 0 & 0 & 0 & 1 \end{bmatrix}, \\ &= \begin{bmatrix} \cos(\phi_i) & -\sin(\phi_i) \cos(\theta_i) & -\sin(\phi_i) \sin(\theta_i) & \sin(\phi_i) [a_i + l_i \sin(\theta_i)] \\ \sin(\phi_i) & \cos(\phi_i) \cos(\theta_i) & \cos(\phi_i) \sin(\theta_i) & -\cos(\phi_i) [a_i + l_i \sin(\theta_i)] \\ 0 & -\sin(\theta_i) & \cos(\theta_i) & -l_i \cos(\theta_i) \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned} \quad (2.7)$$

por lo tanto, la posición $\vec{r}_{i/0}^0 \in \mathbb{R}^{3 \times 1}$ de cada cadena cinemática $\{a_i, l_i\}$ es

$$\vec{r}_{i/0}^0 = \begin{bmatrix} \sin(\phi_i) [a_i + l_i \sin(\theta_i)] \\ -\cos(\phi_i) [a_i + l_i \sin(\theta_i)] \\ -l_i \cos(\theta_i) \end{bmatrix}. \quad (2.8)$$

2.2. Relación Geométrica

Con los resultados de la sección anterior, es posible definir la relación entre cada cadena cinemática y el extremo del robot. Para ello se debe establecer que la posición $\vec{r}_{i/0}^0$ puede ser utilizada para conocer la postura del centro instantáneo de rotación $\mathbf{C}_{v_i/0}^0 \in \mathbb{R}^{4 \times 4}$ a través de la longitud del efector final $p_i \in \mathbb{R}$, vista desde el extremo del eslabón virtual v_i :

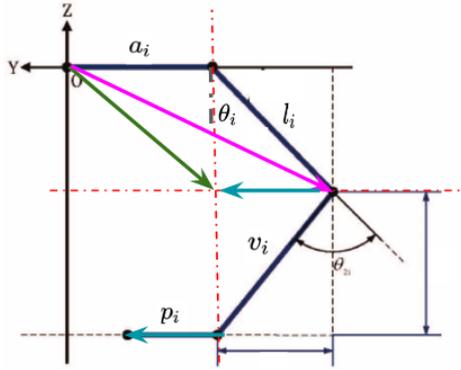


Figura 2.3: Centro Instantáneo de Rotación entre la i -ésima cadena cinemática y el efector final

En la figura anterior se observa que p_i es un vector de traslación sobre el eje y inercial, es decir, sin importar la orientación del cuerpo rígido l_i , este vector siempre se mantendrá alineado con el marco inercial, por lo tanto, su descripción matemática es

$$\begin{aligned} P_i &= R_{\theta_i, x} T_{p_i, y} \\ &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\theta_i) & -\sin(\theta_i) & p_i \cos(\theta_i) \\ 0 & \sin(\theta_i) & \cos(\theta_i) & p_i \sin(\theta_i) \\ 0 & 0 & 0 & 1 \end{bmatrix}. \end{aligned} \quad (2.9)$$

A diferencia de la ecuación (2.5), la matriz de rotación $R_{\theta_i, x}$ elimina el giro del eslabón l_i para garantizar que la traslación p_i sea paralela al eje y inercial. Entonces, la postura del centro instantáneo de rotación se define de la siguiente manera

$$C_{v_i/0}^0 = H_{i/0}^0 P_i, \quad (2.10)$$

por lo tanto, si la ecuación anterior se expande, se obtiene el siguiente resultado

$$\begin{aligned} C_{v_i/0}^0 &= H_{i/0}^0 P_i \\ &= \begin{bmatrix} \cos(\phi_i) & -\sin(\phi_i) & 0 & \sin(\phi_i) [a_i - p_i + l_i \sin(\theta_i)] \\ \sin(\phi_i) & \cos(\phi_i) & 0 & -\cos(\phi_i) [a_i - p_i + l_i \sin(\theta_i)] \\ 0 & 0 & 1 & -l_i \cos(\theta_i) \\ 0 & 0 & 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} \cos(\phi_i) & -\sin(\phi_i) & 0 & c_{x_i/0}^0 \\ \sin(\phi_i) & \cos(\phi_i) & 0 & c_{y_i/0}^0 \\ 0 & 0 & 1 & c_{z_i/0}^0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \end{aligned} \quad (2.11)$$

La posición del centro instantáneo de rotación, o la cuarta columna de la matriz en la ecuación anterior, define el punto común entre todos los elementos que conforman la i -ésima cadena cinemática y el efector final, sin embargo, ahora es necesario describir la conexión entre estos elementos a través del eslabón virtual v_i .

2.3. Cinemática Virtual

La cinemática virtual se refiere al movimiento del eslabón virtual v_i de la figura 2.1 que relaciona a la cadena cinemática $\{a_i, l_i\}$ con el efector final p_i . Debido a que este eslabón relaciona los mencionados anteriormente, el centro instantáneo de rotación se considera el punto de referencia de sus movimientos, los cuales se pueden acotar en una esfera, en el espacio euclidiano, tal como se muestra a continuación

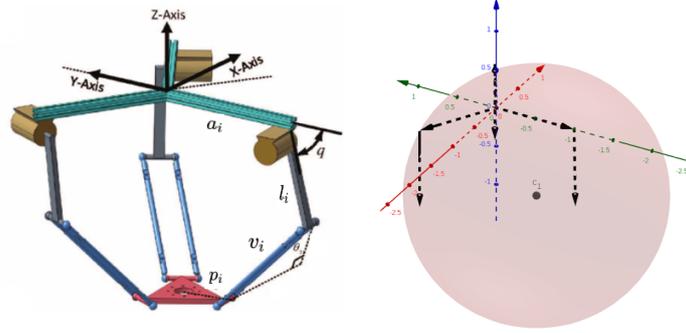


Figura 2.4: Representación de los movimientos de la cadena $\{a_i, l_i\}$ con el efector final p_i a través del eslabón v_i , acotados en una esfera

Esta esfera se define, en coordenadas cartesianas, con la siguiente ecuación

$$\left(x - c_{x_{i/0}}^0\right)^2 + \left(y - c_{y_{i/0}}^0\right)^2 + \left(z - c_{z_{i/0}}^0\right)^2 = v_i^2, \quad (2.12)$$

donde $c_{x_{i/0}}^0, c_{y_{i/0}}^0, c_{z_{i/0}}^0 \in \mathbb{R}$ son las coordenadas en (x, y, z) de cada centro instantáneo de rotación, que pueden obtenerse de la ecuación (2.11). Para el análisis posterior, este resultado puede expandirse de la siguiente manera:

$$\begin{aligned} x^2 + y^2 + z^2 - 2\left(c_{x_{i/0}}^0 x + c_{y_{i/0}}^0 y + c_{z_{i/0}}^0 z\right) + c_{x_{i/0}}^2 + c_{y_{i/0}}^2 + c_{z_{i/0}}^2 &= v_i^2 \\ x^2 + y^2 + z^2 - 2\left(c_{x_{i/0}}^0 x + c_{y_{i/0}}^0 y + c_{z_{i/0}}^0 z\right) + w_i &= v_i^2 \end{aligned} \quad (2.13)$$

Este último resultado encapsula el comportamiento de $\{a_i, l_i\}$ y v_i en una sola esfera, lo que permite que el análisis de todos los cuerpos en el robot se reduzca a calcular el punto donde todas las esferas se intersecan, como se muestra en la siguiente figura

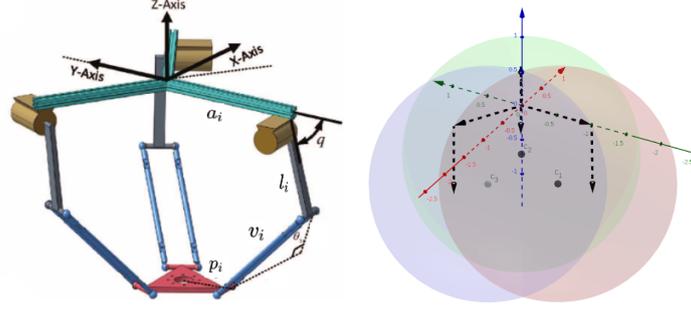


Figura 2.5: Representación del acotamiento en los movimientos de los elementos que conforman un robot paralelo con tres articulaciones.

2.4. Solución para un Sistema con Tres Articulaciones

A partir de aquí, el análisis se debe particularizar para encontrar una solución específica de acuerdo con el número de articulaciones $\bar{\theta} \in \mathbb{R}^{n \times 1}$ en el sistema:

$$\bar{\theta} = [\theta_1 \quad \dots \quad \theta_n]^T \quad (2.14)$$

por esta razón se propone estudiar un robot paralelo con tres articulaciones, separadas angularmente $\phi_i = 120 \text{ deg} = \frac{2}{3}\pi \text{ rad}$, una respecto a la otra. Utilizando las ecuaciones (2.11) y (2.13) y la figura 2.3 como referencia, el sistema propuesto puede definirse matemáticamente como sigue:

- Articulación $i = 1$ ($\phi_1 = 0 \text{ rad}$):

$$\begin{aligned} C_{v_1/0}^0 &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & p_1 - a_1 - l_1 \sin(\theta_1) \\ 0 & 0 & 1 & -l_1 \cos(\theta_1) \\ 0 & 0 & 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} 1 & 0 & 0 & c_{x_1/0}^0 \\ 0 & 1 & 0 & c_{y_1/0}^0 \\ 0 & 0 & 1 & c_{z_1/0}^0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned} \quad (2.15)$$

$$\begin{aligned} x^2 + y^2 + z^2 - 2 \left(c_{x_1/0}^0 x + c_{y_1/0}^0 y + c_{z_1/0}^0 z \right) + c_{x_1/0}^2 + c_{y_1/0}^2 + c_{z_1/0}^2 &= v_1^2 \\ x^2 + y^2 + z^2 - 2 \left([p_1 - a_1 - l_1 \sin(\theta_1)] y - l_1 \cos(\theta_1) z \right) + w_1 &= v_1^2 \end{aligned} \quad (2.16)$$

$$\begin{aligned} w_1 &= c_{x_1/0}^2 + c_{y_1/0}^2 + c_{z_1/0}^2 \\ w_1 &= a_1^2 + l_1^2 + p_1^2 + 2[(a_1 - p_1) l_1 \sin(\theta_1) - a_1 p_1] \end{aligned} \quad (2.17)$$

- Articulación $i = 2$ ($\phi_2 = \frac{2}{3}\pi \text{ rad}$):

$$C_{v_2/0}^0 = \begin{bmatrix} -\frac{1}{2} & -\frac{\sqrt{3}}{2} & 0 & \frac{\sqrt{3}}{2} (a_2 + l_2 \sin(\theta_2) - p_2) \\ \frac{\sqrt{3}}{2} & -\frac{1}{2} & 0 & -\frac{1}{2} (a_2 + l_2 \sin(\theta_2) - p_2) \\ 0 & 0 & 1 & -l_2 \cos(\theta_2) \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} -\frac{1}{2} & -\frac{\sqrt{3}}{2} & 0 & c_{x_2/0}^0 \\ \frac{\sqrt{3}}{2} & -\frac{1}{2} & 0 & c_{y_2/0}^0 \\ 0 & 0 & 1 & c_{z_2/0}^0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.18)$$

$$x^2 + y^2 + z^2 - 2 \left(c_{x_{2/0}}^0 x + c_{y_{2/0}}^0 y + c_{z_{2/0}}^0 z \right) + c_{x_{2/0}}^2 + c_{y_{2/0}}^2 + c_{z_{2/0}}^2 = v_2^2$$

$$x^2 + y^2 + z^2 - 2 \left(\frac{\sqrt{3}}{2} (a_2 + l_2 \sin(\theta_2) - p_2) x - \frac{1}{2} (a_2 + l_2 \sin(\theta_2) - p_2) y - l_2 \cos(\theta_2) z \right) + w_2 = v_2^2 \quad (2.19)$$

$$\begin{aligned} w_2 &= c_{x_{2/0}}^2 + c_{y_{2/0}}^2 + c_{z_{2/0}}^2 \\ w_2 &= a_2^2 + l_2^2 + p_2^2 + 2[(a_2 - p_2) l_2 \sin(\theta_2) - a_2 p_2] \end{aligned} \quad (2.20)$$

■ Articulación $i = 3$ ($\phi_3 = \frac{4}{3}\pi$ rad):

$$C_{v_{3/0}}^0 = \begin{bmatrix} -\frac{1}{2} & \frac{\sqrt{3}}{2} & 0 & -\frac{\sqrt{3}}{2} (a_3 + l_3 \sin(\theta_3) - p_3) \\ -\frac{\sqrt{3}}{2} & -\frac{1}{2} & 0 & -\frac{1}{2} (a_3 + l_3 \sin(\theta_3) - p_3) \\ 0 & 0 & 1 & -l_3 \cos(\theta_3) \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} -\frac{1}{2} & \frac{\sqrt{3}}{2} & 0 & c_{x_{2/0}}^0 \\ -\frac{\sqrt{3}}{2} & -\frac{1}{2} & 0 & c_{y_{2/0}}^0 \\ 0 & 0 & 1 & c_{z_{2/0}}^0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.21)$$

$$x^2 + y^2 + z^2 - 2 \left(c_{x_{3/0}}^0 x + c_{y_{3/0}}^0 y + c_{z_{3/0}}^0 z \right) + c_{x_{3/0}}^2 + c_{y_{3/0}}^2 + c_{z_{3/0}}^2 = v_3^2$$

$$x^2 + y^2 + z^2 + 2 \left(\frac{\sqrt{3}}{2} (a_3 + l_3 \sin(\theta_3) - p_3) x + \frac{1}{2} (a_3 + l_3 \sin(\theta_3) - p_3) y + l_3 \cos(\theta_3) z \right) + w_3 = v_3^2 \quad (2.22)$$

$$\begin{aligned} w_3 &= c_{x_{3/0}}^2 + c_{y_{3/0}}^2 + c_{z_{3/0}}^2 \\ w_3 &= a_3^2 + l_3^2 + p_3^2 + 2[(a_3 - p_3) l_3 \sin(\theta_3) - a_3 p_3] \end{aligned} \quad (2.23)$$

Las ecuaciones (2.16), (2.19) y (2.22) representan las esferas con el movimiento de las cadenas cinemáticas $\{a_1, l_1\}$, $\{a_2, l_2\}$, $\{a_3, l_3\}$ y los eslabones virtuales v_1, v_2, v_3 ; debido a la configuración simétrica de este tipo de sistemas, sus dimensiones pueden simplificarse a

$$\begin{aligned} a_1 &= a_2 = a_3 = a \\ l_1 &= l_2 = l_3 = l \\ p_1 &= p_2 = p_3 = p \\ v_1 &= v_2 = v_3 = v \end{aligned} \quad (2.24)$$

Para definir las coordenadas (x, y, z) donde las tres esferas se intersecan, se deben resolver las ecuaciones simultáneamente, sin embargo, debido a la extensión de los resultados anteriores, el procedimiento de simplificación utilizará el resultado de la ecuación (2.13) para generalizar la solución:

$$x^2 + y^2 + z^2 - 2 \left(c_{x_{1/0}}^0 x + c_{y_{1/0}}^0 y + c_{z_{1/0}}^0 z \right) + w_1 = v^2, \quad (2.25)$$

$$x^2 + y^2 + z^2 - 2 \left(c_{x_{2/0}}^0 x + c_{y_{2/0}}^0 y + c_{z_{2/0}}^0 z \right) + w_2 = v^2, \quad (2.26)$$

$$x^2 + y^2 + z^2 - 2 \left(c_{x_{3/0}}^0 x + c_{y_{3/0}}^0 y + c_{z_{3/0}}^0 z \right) + w_3 = v^2. \quad (2.27)$$

Aunque existen muchas formas para resolver sistemas de tres ecuaciones con tres incógnitas de forma simultánea, se propone utilizar el procedimiento de suma y resta, por lo tanto, las ecuaciones (2.25) y (2.26) se operan de la siguiente manera:

$$\begin{array}{rcl} x^2 + y^2 + z^2 & -2 \left(c_{x_{1/0}}^0 x + c_{y_{1/0}}^0 y + c_{z_{1/0}}^0 z \right) & +w_1 = v^2 \\ -x^2 - y^2 - z^2 & +2 \left(c_{x_{2/0}}^0 x + c_{y_{2/0}}^0 y + c_{z_{2/0}}^0 z \right) & -w_2 = -v^2 \\ \hline +2 \left(c_{x_{2/0}}^0 x + c_{y_{2/0}}^0 y + c_{z_{2/0}}^0 z \right) - 2 \left(c_{x_{1/0}}^0 x + c_{y_{1/0}}^0 y + c_{z_{1/0}}^0 z \right) + w_1 - w_2 = 0 \\ \left(c_{x_{2/0}}^0 - c_{x_{1/0}}^0 \right) x + \left(c_{y_{2/0}}^0 - c_{y_{1/0}}^0 \right) y + \left(c_{z_{2/0}}^0 - c_{z_{1/0}}^0 \right) z = \frac{1}{2} (w_2 - w_1) \end{array} \quad (2.28)$$

Esto también aplica para las ecuaciones (2.25) y (2.27), entonces el procedimiento es:

$$\begin{array}{rcl} x^2 + y^2 + z^2 & -2 \left(c_{x_{1/0}}^0 x + c_{y_{1/0}}^0 y + c_{z_{1/0}}^0 z \right) & +w_1 = v^2 \\ -x^2 - y^2 - z^2 & +2 \left(c_{x_{3/0}}^0 x + c_{y_{3/0}}^0 y + c_{z_{3/0}}^0 z \right) & -w_3 = -v^2 \\ \hline +2 \left(c_{x_{3/0}}^0 x + c_{y_{3/0}}^0 y + c_{z_{3/0}}^0 z \right) - 2 \left(c_{x_{1/0}}^0 x + c_{y_{1/0}}^0 y + c_{z_{1/0}}^0 z \right) + w_1 - w_3 = 0 \\ \left(c_{x_{3/0}}^0 - c_{x_{1/0}}^0 \right) x + \left(c_{y_{3/0}}^0 - c_{y_{1/0}}^0 \right) y + \left(c_{z_{3/0}}^0 - c_{z_{1/0}}^0 \right) z = \frac{1}{2} (w_3 - w_1) \end{array} \quad (2.29)$$

Las ecuaciones (2.28) y (2.29) representan una forma reducida de la intersección entre las esferas del sistema, que son dos planos que también se intersecan:

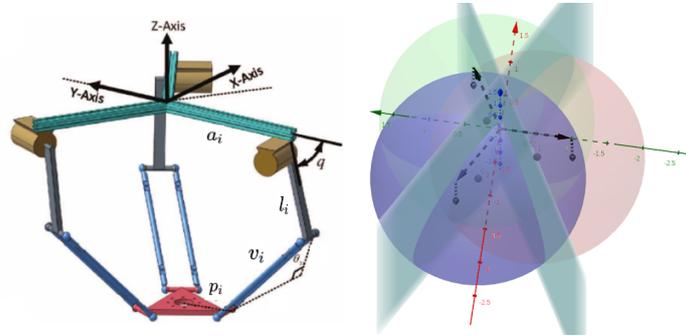


Figura 2.6: Simplificación de las esferas en planos que se intersecan.

Esto significa que se pueden reducir los resultados a una única recta, cuya dirección e intersección con las tres esferas indican la posición del efector final. Una forma intuitiva para conocer la dirección mencionada es conocer los vectores normales a los planos obtenidos:

$$\vec{u} = \begin{bmatrix} c_{x_{2/0}}^0 - c_{x_{1/0}}^0 \\ c_{y_{2/0}}^0 - c_{y_{1/0}}^0 \\ c_{z_{2/0}}^0 - c_{z_{1/0}}^0 \end{bmatrix}, \quad (2.30)$$

$$\vec{v} = \begin{bmatrix} c_{x_{3/0}}^0 - c_{x_{1/0}}^0 \\ c_{y_{3/0}}^0 - c_{y_{1/0}}^0 \\ c_{z_{3/0}}^0 - c_{z_{1/0}}^0 \end{bmatrix}, \quad (2.31)$$

donde $\vec{u}, \vec{v} \in \mathbb{R}^{3 \times 1}$ son los vectores unitarios normales a los planos definidos en las ecuaciones (2.28) y (2.29) (respectivamente). Ambos están relacionados a través de un vector unitario perpendicular a ellos, el cual indica la dirección $\vec{d} \in \mathbb{R}^{3 \times 1}$ donde se encuentra el efector final. La dirección puede calcularse al operar ambos vectores con un producto cruz:

$$\vec{d} = -\vec{u} \times \vec{v} = \vec{v} \times \vec{u} = \begin{bmatrix} d_x \\ d_y \\ d_z \end{bmatrix} \quad (2.32)$$

El signo negativo garantiza que la dirección del efector final será negativa respecto al eje z inercial, tal como se muestra en la figura 2.1. Ahora es necesario calcular las coordenadas (x_0, y_0, z_0) donde esta recta se origina, para posteriormente calcular el punto donde interseca con las esferas. Por simplicidad, las ecuaciones (2.28) y (2.29) pueden utilizarse, además que se establece que el origen sobre el eje z es $z_0 = 0$ (aunque se puede escoger cualquier otro valor numérico):

$$\begin{aligned} (c_{x_{2/0}}^0 - c_{x_{1/0}}^0) x_0 + (c_{y_{2/0}}^0 - c_{y_{1/0}}^0) y_0 &= \frac{1}{2} (w_2 - w_1) \\ (c_{x_{3/0}}^0 - c_{x_{1/0}}^0) x_0 + (c_{y_{3/0}}^0 - c_{y_{1/0}}^0) y_0 &= \frac{1}{2} (w_3 - w_1) \end{aligned} \quad (2.33)$$

Una forma rápida para conocer los valores de x_0 y y_0 es resolver el sistema de ecuaciones, mostrado anteriormente, utilizando matrices:

$$\begin{aligned} \begin{bmatrix} c_{x_{2/0}}^0 - c_{x_{1/0}}^0 & c_{y_{2/0}}^0 - c_{y_{1/0}}^0 \\ c_{x_{3/0}}^0 - c_{x_{1/0}}^0 & c_{y_{3/0}}^0 - c_{y_{1/0}}^0 \end{bmatrix} \begin{bmatrix} x_0 \\ y_0 \end{bmatrix} &= \frac{1}{2} \begin{bmatrix} w_2 - w_1 \\ w_3 - w_1 \end{bmatrix} \\ \begin{bmatrix} x_0 \\ y_0 \end{bmatrix} &= \begin{bmatrix} c_{x_{2/0}}^0 - c_{x_{1/0}}^0 & c_{y_{2/0}}^0 - c_{y_{1/0}}^0 \\ c_{x_{3/0}}^0 - c_{x_{1/0}}^0 & c_{y_{3/0}}^0 - c_{y_{1/0}}^0 \end{bmatrix}^{-1} \left(\frac{1}{2} \begin{bmatrix} w_2 - w_1 \\ w_3 - w_1 \end{bmatrix} \right), \end{aligned} \quad (2.34)$$

estos resultados son el punto de partida para conocer la recta que interseca con las tres esferas del sistema. Primero, y con el resultado de la ecuación (2.32), se define la ecuación de la recta sobre los ejes x y y inerciales, esto es

$$\frac{x - x_0}{d_x} = \frac{y - y_0}{d_y} = \frac{z - z_0}{d_z} \Rightarrow \frac{x - x_0}{d_x} = \frac{z - z_0}{d_z} \ \& \ \frac{y - y_0}{d_y} = \frac{z - z_0}{d_z}$$

$$x(z) = x_0 + (z - z_0) \frac{d_x}{d_z} \quad (2.35)$$

$$y(z) = y_0 + (z - z_0) \frac{d_y}{d_z}, \quad (2.36)$$

ya que estos resultados dependen de la coordenada en z , se puede sustituirlos en cualquiera de las ecuaciones (2.25) a (2.27), en este ejemplo, se escoge la ecuación (2.25) para calcular las coordenadas completamente, además que anteriormente se estableció que $z_0 = 0$:

$$\left(x_0 + z \left(\frac{d_x}{d_z} \right) \right)^2 + \left(y_0 + z \left(\frac{d_y}{d_z} \right) \right)^2 + z^2 - 2 \left(c_{x_{1/0}}^0 x + c_{y_{1/0}}^0 y + c_{z_{1/0}}^0 z \right) + w_1 = v^2,$$

$$\left[\left(\frac{d_x}{d_z} \right)^2 + \left(\frac{d_y}{d_z} \right)^2 + 1 \right] z^2 + 2 \left[(x_0 - c_{x_{1/0}}^0) \left(\frac{d_x}{d_z} \right) + (y_0 - c_{y_{1/0}}^0) \left(\frac{d_y}{d_z} \right) - c_{z_{1/0}}^0 \right] z + x_0^2 + y_0^2 - 2 (c_{x_{1/0}}^0 x_0 + c_{y_{1/0}}^0 y_0) + w_1 - v^2 = 0, \quad (2.37)$$

donde este resultado una ecuación de segundo orden de la forma $az^2 + bz + c = 0$, de la cual se calculan sus raíces y se escoge aquella más negativa, ya que la raíz indica la posición, sobre el eje z , del efector final. Para obtener las coordenadas sobre x y y , se sustituye la raíz más negativa de la ecuación (2.37) en las ecuaciones (2.35) y (2.36), lo que permite obtener la posición del efector final respecto al marco inercial $\vec{r}_{e/0}^0 \in \mathbb{R}^{3 \times 1}$

$$\vec{r}_{e/0}^0 = \begin{bmatrix} x \\ y \\ z \end{bmatrix}. \quad (2.38)$$

2.5. Algoritmo Numérico

Aunque el procedimiento mostrado en la sección anterior es extenso, el algoritmo para calcular la cinemática directa se puede reducir al cálculo de las ecuaciones (2.11), (2.30) a (2.32) y (2.34) a (2.37), cuyo procedimiento se describe en el algoritmo 1 (véase apéndice A.1 para verificar el código en MATLAB).

Algoritmo 1: Cinemática Directa

```

Entrada:  $(\theta, a, l, p, v)$  ; /* Parámetros geométricos */
 $\phi \leftarrow \frac{2}{3}\pi$  ; /* Separación entre articulaciones */
for  $i \leftarrow 1$  to 3 do
   $C_{v_{i/0}}^0 \leftarrow$  ecuación (2.11) ; /*  $i$ -ésimo centro instantáneo de rotación */
   $w_i \leftarrow c_{x_{i/0}}^2 + c_{y_{i/0}}^2 + c_{z_{i/0}}^2$  ; /* Suma de los cuadrados de sus coordenadas */
end
 $\vec{u} \leftarrow$  ecuación (2.30) ; /* Vector unitario normal al plano de la ecuación (2.28) */
 $\vec{v} \leftarrow$  ecuación (2.31) ; /* Vector unitario normal al plano de la ecuación (2.29) */
 $\vec{d} \leftarrow$  ecuación (2.32) ; /* Dirección de la recta que interseca las esferas */
 $\begin{bmatrix} x_0 \\ y_0 \end{bmatrix} \leftarrow$  ecuación (2.34) ; /* Origen de la recta que interseca las esferas */
 $z \leftarrow$  ecuación (2.37) ; /* Coordenada del efector final en  $z$  */
 $x(z) \leftarrow$  ecuación (2.35) ; /* Coordenada del efector final en  $x$  */
 $y(z) \leftarrow$  ecuación (2.36) ; /* Coordenada del efector final en  $y$  */
Salida:  $(x, y, z)$ 

```

2.6. Resultado Numérico de la Cinemática Directa

En esta sección se realiza un ejemplo sobre los cálculos de la cinemática directa de un robot paralelo (figura 1.1) con las características geométricas establecidas en la tabla 1. El cálculo se

realiza utilizando el código mostrado en el apéndice A.1, el cual se basa en el algoritmo 1 y da como resultado que la posición del efector final es

$$\vec{r}_{e/0}^0 = \begin{bmatrix} -0.5661 \\ -0.0522 \\ -1.2180 \end{bmatrix} \quad (2.39)$$

Este resultado se representa, gráficamente, como se muestra en la figura 2.7.



Parámetro	Valor Numérico
θ_1	0.4434 [rad]
θ_2	0.0249 [rad]
θ_3	0.9590 [rad]
a	0.38457769 [metros]
l	0.64 [metros]
p	0.215 [metros]
v	0.94 [metros]

Figura 2.7: Representación gráfica de la Cinemática Directa.

Tabla 1: Parámetros Geométricos del Robot propuesto.

3. Cinemática Inversa

Existen múltiples formas para resolver el problema de la cinemática inversa, cuya premisa es conocer la posición de las articulaciones θ para que el efector final alcance una posición deseada $\vec{r}_d \in \mathbb{R}^{3 \times 1}$. Se propone utilizar una ecuación que garantice la estabilidad del resultado, entonces, tal como se describe en [4], este problema puede resolverse al establecer una ecuación de error $e(t) \in \mathbb{R}^{3 \times 1}$ de la forma

$$e(t) = \vec{r}_d - \vec{r}_{e/0}^0 = \begin{bmatrix} x_d \\ y_d \\ z_d \end{bmatrix} - \begin{bmatrix} x \\ y \\ z \end{bmatrix}. \quad (3.1)$$

El objetivo de utilizar esta ecuación es lograr que el error converja asintóticamente, esto significa que el error debe tender a cero ($e(t) \rightarrow 0$) conforme el tiempo tiende a infinito ($t \rightarrow \infty$), cuya dinámica $\dot{e}(t) \in \mathbb{R}^{3 \times 1}$ se describe, idealmente, como

$$\dot{e}(t) = -Ke(t), \quad (3.2)$$

donde $K \in \mathbb{R}^{3 \times 3}$ es una matriz constante, definida positiva. Para relacionar las ecuaciones (3.1) y (3.3), se deriva la primera en función del tiempo y posteriormente se iguala con la segunda, lo que da el siguiente resultado

$$\begin{aligned} \dot{e}(t) = -Ke(t) &= \dot{\vec{r}}_d - \dot{\vec{r}}_{e/0}^0 \\ -Ke(t) &= -\dot{\vec{r}}_{e/0}^0 \\ Ke(t) &= \dot{\vec{r}}_{e/0}^0. \end{aligned} \quad (3.3)$$

En [1] se establece la forma diferencial de la cinemática del efector final, que se describe gracias a la matriz jacobiana del sistema, esto es

$$\dot{\bar{r}}_{e/0}^0 = J^I(\bar{\theta}) \dot{\bar{\theta}} \quad (3.4)$$

con $\bar{\theta}, \dot{\bar{\theta}} \in \mathbb{R}^{n \times 1}$, que representan las coordenadas generalizadas del sistema, es decir, la posición y velocidad instantáneas de las articulaciones del robot. La matriz jacobiana **analítica** $J^I(\bar{\theta}) \in \mathbb{R}^{3 \times n}$ se obtiene al derivar las ecuaciones (2.35) a (2.37) respecto a la posición de las articulaciones $\bar{\theta}$

$$J^I(\bar{\theta}) = \frac{\partial \bar{r}_{e/0}^0}{\partial \bar{\theta}} = \begin{bmatrix} \frac{\partial x}{\partial \theta_1} & \cdots & \frac{\partial x}{\partial \theta_n} \\ \frac{\partial y}{\partial \theta_1} & \cdots & \frac{\partial y}{\partial \theta_n} \\ \frac{\partial z}{\partial \theta_1} & \cdots & \frac{\partial z}{\partial \theta_n} \end{bmatrix}. \quad (3.5)$$

Aunque este es el procedimiento más conocido para calcular la matriz, puede ser imprecisa si no se escoge un tamaño de paso lo suficientemente pequeño para obtenerla numéricamente, además, debido a la complejidad para calcular las derivadas de cada coordenada cartesiana respecto a cada una de las articulaciones, se propone calcularla una forma **geométrica**, tal como ocurre con los robots seriales [5].

3.1. Matriz Jacobiana Geométrica

La relación entre las articulaciones y el efector final se da a través de los elementos que construyen cada cadena cinemática de un robot paralelo:

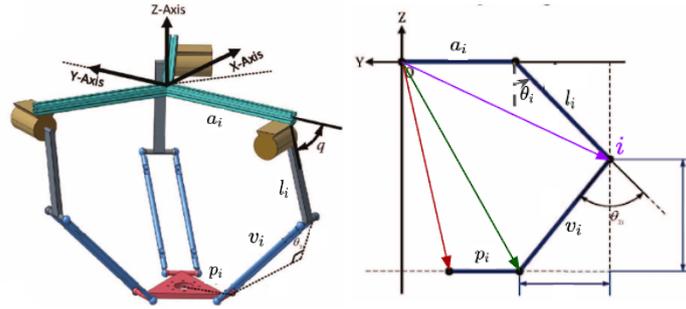


Figura 3.1: Vectores de posición de las uniones entre cuerpos rígidos en un Robot Delta

En secciones previas se calculó la cinemática directa hasta el i -ésimo marco móvil, cuyo resultado se muestra en la ecuación (2.8), asimismo, la cinemática directa hasta el extremo del robot se calcula con el algoritmo mostrado en la sección anterior. Entonces, estos vectores pueden definirse de la siguiente manera:

$$\bar{r}_{i/0}^0 = \begin{bmatrix} \sin(\phi_i) [a_i + l_i \sin(\theta_i)] \\ -\cos(\phi_i) [a_i + l_i \sin(\theta_i)] \\ -l_i \cos(\theta_i) \end{bmatrix} \quad (3.6)$$

$$\bar{r}_{e/0}^0 = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad (\text{calculado con el algoritmo 1}) \quad (3.7)$$

Para conocer la relación de estos vectores con el faltante ($\bar{r}_{p_i/0}^0$), se propone hacer una suma vectorial

$$\bar{r}_{e/0}^0 + \bar{r}_{p_i/e}^0 = \bar{r}_{i/0}^0 + \bar{r}_{p_i/i}^0, \quad (3.8)$$

que sin importar los valores de $\vec{r}_{p_i/e}^0$ y $\vec{r}_{v_i/i}^0$ (por ahora), se puede derivar la ecuación anterior en función del tiempo para conocer la velocidad del efector final:

$$\dot{\vec{r}}_{e/0}^0 + \dot{\vec{r}}_{p_i/e}^0 = \dot{\vec{r}}_{i/0}^0 + \dot{\vec{r}}_{p_i/i}^0, \quad (3.9)$$

donde las velocidades de los marcos móviles $\vec{r}_{p_i/e}^0$, $\vec{r}_{i/0}^0$ y $\vec{r}_{p_i/i}^0$ pueden definirse gracias al *transport theorem* [6] como

$$\dot{\vec{r}}_{e/0}^0 + \omega_{p_i/e}^0 \times \vec{r}_{p_i/e}^0 = \dot{\vec{r}}_{i/0}^0 + \omega_{i/a_i}^0 \times \vec{r}_{i/a_i}^0 + \omega_{p_i/i}^0 \times \vec{r}_{p_i/i}^0, \quad (3.10)$$

sin embargo, ya que el efector final solo se desplaza linealmente, la velocidad angular $\omega_{p_i/e}^0$ es igual a cero. Por otro lado, la velocidad del cuerpo rígido a_i (definida con el término $\dot{\vec{r}}_{a_i/0}^0$) se considera igual a cero, ya que la base del robot no se desplaza, por lo tanto, se puede simplificar la ecuación anterior de la siguiente manera:

$$\begin{aligned} \dot{\vec{r}}_{e/0}^0 &= \dot{\vec{r}}_{i/0}^0 + \omega_{i/a_i}^0 \times \vec{r}_{i/a_i}^0 + \omega_{p_i/i}^0 \times \vec{r}_{p_i/i}^0 \\ &= \left[\left(R_{\phi_i,z} \vec{n}_{a_i/0}^0 \right) \times \vec{r}_{i/a_i}^0 \right] \dot{\theta}_i + \omega_{p_i/i}^0 \times \vec{r}_{p_i/i}^0, \end{aligned} \quad (3.11)$$

siendo $R_{\phi_i,z} \in \mathbb{R}^{3 \times 3}$ una matriz de rotación sobre el eje z inercial, descrita como matriz de transformación homogénea en la ecuación (2.6). Asimismo, $\vec{n}_{a_i/0}^0 \in \mathbb{R}^{3 \times 1}$ es el eje de acción de la i -ésima articulación, el cual puede definirse como

$$\vec{n}_{a_i/0}^0 = \begin{bmatrix} -1 \\ 0 \\ 0 \end{bmatrix}, \quad (3.12)$$

$$R_{\phi_i,z} \vec{n}_{a_i/0}^0 = \begin{bmatrix} -\cos(\phi_i) \\ -\sin(\phi_i) \\ 0 \end{bmatrix}, \quad (3.13)$$

donde el signo negativo indica el sentido de giro de la articulación. La velocidad angular del eslabón virtual $\omega_{p_i/i}^0$ puede calcularse, sin embargo, depende del resto de los eslabones virtuales y articulaciones en el robot, esto la haría una ecuación redundante, por lo que se puede eliminar el término al operar su producto cruz con un *triple producto escalar* [7] tal como se muestra a continuación

$$\begin{aligned} \vec{r}_{p_i/i}^{T} \dot{\vec{r}}_{e/0}^0 &= \vec{r}_{p_i/i}^{T} \left[\left(R_{\phi_i,z} \vec{n}_{a_i/0}^0 \right) \times \vec{r}_{i/a_i}^0 \right] \dot{\theta}_i + \vec{r}_{p_i/i}^{T} \left(\omega_{p_i/i}^0 \times \vec{r}_{p_i/i}^0 \right) \\ &= \vec{r}_{p_i/i}^{T} \left[\left(R_{\phi_i,z} \vec{n}_{a_i/0}^0 \right) \times \vec{r}_{i/a_i}^0 \right] \dot{\theta}_i + \underbrace{\omega_{p_i/i}^T \left(\vec{r}_{p_i/i}^0 \times \vec{r}_{p_i/i}^0 \right)}_{=0_{3 \times 1}} \\ &= \left(R_{\phi_i,z} \vec{n}_{a_i/0}^0 \right)^T \left[\vec{r}_{i/a_i}^0 \times \vec{r}_{p_i/i}^0 \right] \dot{\theta}_i \end{aligned} \quad (3.14)$$

que es una ecuación válida ya que el producto cruz entre dos vectores iguales, sin importar sus valores, siempre será igual a cero. Del resultado anterior se pueden definir los vectores \vec{r}_{i/a_i}^0 y $\vec{r}_{p_i/i}^{T}$

como

$$\begin{aligned}
\vec{r}_{i/a_i}^0 &= \vec{r}_{i/0}^0 - R_{\phi_i, z} T_{-a_i, y} \\
&= \begin{bmatrix} \sin(\phi_i) [a_i + l_i \sin(\theta_i)] \\ -\cos(\phi_i) [a_i + l_i \sin(\theta_i)] \\ -l_i \cos(\theta_i) \end{bmatrix} - \begin{bmatrix} a_i \sin(\phi_i) \\ -a_i \cos(\phi_i) \\ 0 \end{bmatrix} \\
&= \begin{bmatrix} l_i \sin(\phi_i) \sin(\theta_i) \\ -l_i \cos(\phi_i) \sin(\theta_i) \\ -l_i \cos(\theta_i) \end{bmatrix},
\end{aligned} \tag{3.15}$$

$$\begin{aligned}
\vec{r}_{p_i/i}^0 &= \vec{r}_{p_i/0}^0 - \vec{r}_{i/0}^0 \\
&= \vec{r}_{e/0}^0 - R_{\phi_i, z} T_{p_i, y} - \vec{r}_{i/0}^0 \\
&= \begin{bmatrix} x \\ y \\ z \end{bmatrix} - \begin{bmatrix} -p_i \sin(\phi_i) \\ p_i \cos(\phi_i) \\ 0 \end{bmatrix} - \begin{bmatrix} \sin(\phi_i) [a_i + l_i \sin(\theta_i)] \\ -\cos(\phi_i) [a_i + l_i \sin(\theta_i)] \\ -l_i \cos(\theta_i) \end{bmatrix} \\
&= \begin{bmatrix} x - \sin(\phi_i) [a_i - p_i + l_i \sin(\theta_i)] \\ y + \cos(\phi_i) [a_i - p_i + l_i \sin(\theta_i)] \\ z + l_i \cos(\theta_i) \end{bmatrix}.
\end{aligned} \tag{3.16}$$

Si bien las ecuaciones (3.12), (3.15) y (3.16) pueden operarse y sustituirse en la ecuación (3.14), esto daría como resultado una ecuación que solo incluiría a la i -ésima articulación, es decir, solo incluiría los efectos de una articulación en vez de incluir a las tres al mismo tiempo. Para esto, debemos definir las ecuaciones para cada una de la siguiente manera:

$$\vec{r}_{p_1/1}^T \dot{\vec{r}}_{e/0}^0 = \left(R_{0, z} \vec{n}_{a_1/0}^0 \right)^T \left[\vec{r}_{1/a_1}^0 \times \vec{r}_{p_1/1}^0 \right] \dot{\theta}_1 \tag{3.17}$$

$$\vec{r}_{p_2/2}^T \dot{\vec{r}}_{e/0}^0 = \left(R_{\frac{2}{3}\pi, z} \vec{n}_{a_2/0}^0 \right)^T \left[\vec{r}_{2/a_2}^0 \times \vec{r}_{p_2/2}^0 \right] \dot{\theta}_2 \tag{3.18}$$

$$\vec{r}_{p_3/3}^T \dot{\vec{r}}_{e/0}^0 = \left(R_{\frac{4}{3}\pi, z} \vec{n}_{a_3/0}^0 \right)^T \left[\vec{r}_{3/a_3}^0 \times \vec{r}_{p_3/3}^0 \right] \dot{\theta}_3 \tag{3.19}$$

Ya que las tres ecuaciones tienen términos comunes ($\dot{\vec{r}}_{e/0}^0$) o que pertenecen al vector de coordenadas generalizadas $\dot{\theta} \in \mathbb{R}^{3 \times 1}$, se pueden agrupar en una sola ecuación tal que

$$\begin{bmatrix} \vec{r}_{p_1/1}^0 & \vec{r}_{p_2/2}^0 & \vec{r}_{p_3/3}^0 \end{bmatrix}^T \dot{\vec{r}}_{e/0}^0 = \begin{bmatrix} \left(R_{0, z} \vec{n}_{a_1/0}^0 \right)^T \left[\vec{r}_{1/a_1}^0 \times \vec{r}_{p_1/1}^0 \right] & 0 & 0 \\ 0 & \left(R_{\frac{2}{3}\pi, z} \vec{n}_{a_2/0}^0 \right)^T \left[\vec{r}_{2/a_2}^0 \times \vec{r}_{p_2/2}^0 \right] & 0 \\ 0 & 0 & \left(R_{\frac{4}{3}\pi, z} \vec{n}_{a_3/0}^0 \right)^T \left[\vec{r}_{3/a_3}^0 \times \vec{r}_{p_3/3}^0 \right] \end{bmatrix} \dot{\theta} \tag{3.20}$$

Si los términos a la izquierda y derecha de la ecuación anterior se agrupan, se puede reducir el resultado a

$$\dot{\vec{r}}_{e/0}^0 = J^I(\vec{n}, \vec{r}) \dot{\theta}, \tag{3.21}$$

donde $J^I(\vec{n}, \vec{r})$ representa a la matriz jacobiana **geométrica**, la cual está definida de la siguiente manera

$$J^I(\vec{n}, \vec{r}) = J_{\vec{r}}^\dagger J_{\vec{\theta}} \tag{3.22}$$

siendo $J_{\vec{r}} \in \mathbb{R}^{3 \times 3}$ la sub-matriz jacobiana que define los efectos lineales en el extremo del robot, mientras que $J_{\vec{\theta}} \in \mathbb{R}^{3 \times 3}$ establece la relación entre el efector final y los movimientos de las articulaciones; ambas sub-matrices se definen como

$$J_{\vec{r}}^I(\vec{n}, \vec{r}) = \begin{bmatrix} \vec{r}_{p_1/1}^0 & \vec{r}_{p_2/2}^0 & \vec{r}_{p_3/3}^0 \end{bmatrix}^T, \quad (3.23)$$

$$J_{\vec{\theta}}^I(\vec{n}, \vec{r}) = \begin{bmatrix} \left(R_{0,z} \vec{n}_{a_1/0}^0 \right)^T \begin{bmatrix} \vec{r}_{1/a_1}^0 \times \vec{r}_{p_1/1}^0 \\ 0 \\ 0 \end{bmatrix} & 0 & 0 \\ 0 & \left(R_{\frac{2}{3}\pi,z} \vec{n}_{a_2/0}^0 \right)^T \begin{bmatrix} \vec{r}_{2/a_2}^0 \times \vec{r}_{p_2/2}^0 \\ 0 \\ 0 \end{bmatrix} & 0 \\ 0 & 0 & \left(R_{\frac{4}{3}\pi,z} \vec{n}_{a_3/0}^0 \right)^T \begin{bmatrix} \vec{r}_{3/a_3}^0 \times \vec{r}_{p_3/3}^0 \end{bmatrix} \end{bmatrix}, \quad (3.24)$$

donde cada término dentro de la matriz se calcula directamente con las ecuaciones (3.15) y (3.16), o realizando un procedimiento numérico al restar los vectores mostrados en ellas. La forma pseudo-inversa de la ecuación (3.23) se puede simplificar tal como se muestra en [8]:

$$J_{\vec{r}}^\dagger(\vec{n}, \vec{r}) = \frac{1}{\vec{r}_{p_1/1}^{\vec{r}^T} \left(\vec{r}_{p_2/2}^0 \times \vec{r}_{p_3/3}^0 \right)} \begin{bmatrix} \vec{r}_{p_2/2}^0 \times \vec{r}_{p_3/3}^0 & \vec{r}_{p_3/3}^0 \times \vec{r}_{p_1/1}^0 & \vec{r}_{p_1/1}^0 \times \vec{r}_{p_2/2}^0 \end{bmatrix}, \quad (3.25)$$

esto significa que la matriz jacobiana en la ecuación (3.22) se puede reducir a

$$J^I(\vec{n}, \vec{r}) = \frac{1}{\vec{r}_{p_1/1}^{\vec{r}^T} \left(\vec{r}_{p_2/2}^0 \times \vec{r}_{p_3/3}^0 \right)} \left[\left(\vec{r}_{p_2/2}^0 \times \vec{r}_{p_3/3}^0 \right) \left(R_{0,z} \vec{n}_{a_1/0}^0 \right)^T \begin{bmatrix} \vec{r}_{1/a_1}^0 \times \vec{r}_{p_1/1}^0 \\ 0 \\ 0 \end{bmatrix} + \left(\vec{r}_{p_3/3}^0 \times \vec{r}_{p_1/1}^0 \right) \left(R_{\frac{2}{3}\pi,z} \vec{n}_{a_2/0}^0 \right)^T \begin{bmatrix} \vec{r}_{2/a_2}^0 \times \vec{r}_{p_2/2}^0 \\ 0 \\ 0 \end{bmatrix} + \left(\vec{r}_{p_1/1}^0 \times \vec{r}_{p_2/2}^0 \right) \left(R_{\frac{4}{3}\pi,z} \vec{n}_{a_3/0}^0 \right)^T \begin{bmatrix} \vec{r}_{3/a_3}^0 \times \vec{r}_{p_3/3}^0 \end{bmatrix} \right]. \quad (3.26)$$

Cualquiera de las dos formas para calcular la matriz jacobiana es válida, sin embargo, aquí se desarrolló el código en MATLAB, ejemplificado en el algoritmo 2, para la ecuación (3.22) (apéndice A.2). Se debe destacar que es un algoritmo muy simple ya que requiere pocas líneas de código y trabaja con todos los parámetros geométricos del sistema, además que todos sus términos se obtienen del cálculo de la cinemática directa.

Algoritmo 2: Matriz Jacobiana Geométrica

```

Entrada:  $\left( \vec{r}_{e/0}^0, \vec{\theta}, a, l, p \right)$ ; /* Parámetros geométricos */
 $\phi \leftarrow \frac{2}{3}\pi$ ; /* Separación entre articulaciones */
 $J_{\vec{r}} \leftarrow \text{zeros}(3, 3)$ 
 $J_{\vec{\theta}} \leftarrow \text{zeros}(3, 3)$ ; /* Inicialización de las matrices jacobianas */
for  $i \leftarrow 1$  to 3 do
     $R_{\phi_i,z} \vec{n}_{a_i/0}^0 \leftarrow \begin{bmatrix} -\cos(\phi_i) \\ -\sin(\phi_i) \\ 0 \end{bmatrix}$ ; /* Eje de acción de la  $i$ -ésima articulación */
     $\vec{r}_{p_i/i}^0 \leftarrow$  ecuación (3.16)
     $\vec{r}_{i/a_i}^0 \leftarrow$  ecuación (3.15)
     $J_{\vec{r}}(i, :) \leftarrow \vec{r}_{p_i/i}^{\vec{r}^T}$ 
     $J_{\vec{\theta}}(i, i) \leftarrow \left( R_{\phi_i,z} \vec{n}_{a_i/0}^0 \right)^T \begin{bmatrix} \vec{r}_{i/a_i}^0 \times \vec{r}_{p_i/i}^0 \end{bmatrix}$ 
end
 $J^I(\vec{n}, \vec{r}) \leftarrow J_{\vec{r}}^\dagger J_{\vec{\theta}}$ ; /* Matriz Jacobiana Geométrica de la ecuación (3.22) */
Salida:  $J(\vec{n}, \vec{r})$ 

```

3.2. Forma Numérica de la Solución

La ecuación (3.21) se sustituye en la ecuación (3.3), lo que proporciona el siguiente resultado:

$$\begin{aligned} J^I(\vec{n}, \vec{r}) \dot{\bar{\theta}} &= \text{Ke}(t) \\ J^I(\vec{n}, \vec{r})^\dagger J^I(\vec{n}, \vec{r}) \dot{\bar{\theta}} &= J^I(\vec{n}, \vec{r})^\dagger \text{Ke}(t) \\ \dot{\bar{\theta}} &= J^I(\vec{n}, \vec{r})^\dagger \text{Ke}(t). \end{aligned} \quad (3.27)$$

De acuerdo con [9], el producto $J^I(\vec{n}, \vec{r})^\dagger J^I(\vec{n}, \vec{r}) = \mathbb{I} \in \mathbb{R}^{3 \times 3}$ es válido si el número de filas en la matriz jacobiana es mayor al número de columnas en la misma. Esto significa que es posible calcular la cinemática inversa si, y solo si, el número de grados de libertad en el efector final (i.e. el número de coordenadas en que puede desplazarse) es mayor o igual al número de articulaciones en el sistema. Si el robot tuviera más de tres articulaciones, pero el efector final solo pudiera desplazarse linealmente sobre (x, y, z) , la solución propuesta no funcionaría. Por lo tanto se puede utilizarlo para el robot propuesto en el apartado 2.4 al integrar la ecuación anterior como sigue

$$\begin{aligned} \int_{\bar{\theta}(0)}^{\bar{\theta}(t)} \dot{\bar{\theta}}(\tau) d\tau &= \int_0^t J^I(\vec{n}, \vec{r})^\dagger \text{Ke}(\tau) d\tau \\ \bar{\theta}(t) &= \bar{\theta}(0) + \int_0^t J^I(\vec{n}, \vec{r})^\dagger \text{Ke}(\tau) d\tau, \end{aligned} \quad (3.28)$$

La integral en el resultado anterior puede calcularse analíticamente al definir las ecuaciones que constituyen a la matriz jacobiana $J^I(\vec{n}, \vec{r})$ y al error $e(t)$, sin embargo, también es posible calcularla de forma numérica con el algoritmo 3.

Algoritmo 3: Cinemática Inversa

```

Entrada:  $(\vec{r}_d, \bar{\theta}(0), a, l, p, v, \Delta t)$ ; /* Parámetros geométricos */
 $\bar{\theta}(t) \leftarrow \bar{\theta}(0)$ ; /* Condiciones iniciales de la integración numérica */
for  $k \leftarrow 1$  to 1500 do
     $\vec{r}_{e/0}^0(\bar{\theta}_k) \leftarrow$  algoritmo 1; /* Cinemática Directa */
     $e(k) \leftarrow$  ecuación (3.1); /* Cálculo del Error */
    if  $\max(|e(k)|) < 0.001$  then
        break; /* Si el error es menor a 0.001, se detiene la iteración */
    else
         $J(\bar{\theta}_k) \leftarrow$  algoritmo 2; /* Si no, calcula la matriz jacobiana numérica */
         $\bar{\theta}(k+1) \leftarrow \bar{\theta}(k) + [J(\bar{\theta}_k)^\dagger \text{Ke}(k)] \Delta t$ ; /* Forma numérica: ecuación (3.28) */
    end
end
Salida:  $\bar{\theta}(t)$ 

```

3.3. Resultados

En esta sección se realiza un ejemplo sobre los cálculos de la cinemática inversa de un robot paralelo (figura 1.1) con las características geométricas establecidas en la tabla 1. El cálculo se

realiza utilizando el código mostrado en el apéndice A.3, el cual se basa en el algoritmo 3. Para ello se propone que la posición inicial de las articulaciones $\bar{\theta}$ sea

$$\bar{\theta}(0) = [0 \quad 0 \quad 0]^T. \quad (3.29)$$

A partir de esto, se propone que el efector final alcance una posición deseada, la cual fue calculada en la sección anterior y los resultados se muestran en la ecuación (2.39). Además, el tamaño de paso para la integración numérica (ecuación (3.28) y algoritmo 3) se define con $\Delta t = 0.001$. Entonces, el resultado del cálculo se representa, gráficamente, como se muestra en la siguiente figura.



Figura 3.2: Representación gráfica de la Cinemática Inversa.

4. Cinemática Diferencial

En la sección anterior se estableció que la ecuación (3.4) es la definición de la velocidad instantánea del efector final, que tal como se describe en [10], esta es la cinemática diferencial, que se encarga de estudiar la velocidad y aceleración instantáneas de un sistema sin considerar factores como la geometría del cuerpo y la distribución de su masa. A partir de dicha ecuación, se puede describir la velocidad instantánea del efector final de un robot paralelo de la siguiente manera

$$\dot{\vec{r}}_{e/0}^0 = \begin{bmatrix} v_{p_x/0}^0 \\ v_{p_y/0}^0 \\ v_{p_z/0}^0 \end{bmatrix} = J^I(\vec{n}, \vec{r}) \dot{\bar{\theta}}, \quad (4.1)$$

siendo $v_{p_x/0}^0, v_{p_y/0}^0, v_{p_z/0}^0 \in \mathbb{R}$ las velocidades instantáneas lineales, respecto al marco inercial, sobre los ejes (x, y, z) del efector final. Asimismo, su aceleración lineal instantánea se define al derivar, en función del tiempo, el resultado anterior

$$\ddot{\vec{r}}_{e/0}^0 = \begin{bmatrix} \dot{v}_{p_x/0}^0 \\ \dot{v}_{p_y/0}^0 \\ \dot{v}_{p_z/0}^0 \end{bmatrix} = \dot{J}^I(\vec{n}, \vec{r}) \dot{\bar{\theta}} + J^I(\vec{n}, \vec{r}) \ddot{\bar{\theta}}, \quad (4.2)$$

donde el término $\dot{J}^I(\vec{n}, \vec{r}) \in \mathbb{R}^{3 \times n}$ puede calcularse al realizar una expansión diádica como se muestra en [11]:

$$\dot{J}^I(\vec{n}, \vec{r}) = \sum_{i=1}^n \left[\frac{\partial J^I(\cdot, i)}{\partial \bar{\theta}} \cdot \dot{\theta}_i \right], \quad (4.3)$$

este resultado representa una forma sencilla de derivar la matriz jacobiana, columna por columna, sin necesidad de obtener una matriz tridimensional.

4.1. Derivada de la Matriz Jacobiana Geométrica

Ya que existe una forma geométrica de la matriz jacobiana, definida en la ecuación (3.22), es posible derivar sus términos en función del tiempo para obtener los valores de $J^I(\vec{n}, \vec{r})$. Partiendo de la ecuación (3.14), sus términos se operan como se muestra a continuación

$$\begin{aligned} \dot{\vec{r}}_{p_i/i}^T \dot{\vec{r}}_{e/0}^0 + \vec{r}_{p_i/i}^T \ddot{\vec{r}}_{e/0}^0 &= \frac{d}{dt} \left(R_{\phi_i, z} \vec{n}_{a_i/0}^0 \right)^T \left[\vec{r}_{i/a_i}^0 \times \vec{r}_{p_i/i}^0 \right] \dot{\theta}_i + \left(R_{\phi_i, z} \vec{n}_{a_i/0}^0 \right)^T \left[\dot{\vec{r}}_{i/a_i}^0 \times \vec{r}_{p_i/i}^0 \right] \dot{\theta}_i \\ &+ \left(R_{\phi_i, z} \vec{n}_{a_i/0}^0 \right)^T \left[\vec{r}_{i/a_i}^0 \times \dot{\vec{r}}_{p_i/i}^0 \right] \dot{\theta}_i + \left(R_{\phi_i, z} \vec{n}_{a_i/0}^0 \right)^T \left[\vec{r}_{i/a_i}^0 \times \vec{r}_{p_i/i}^0 \right] \ddot{\theta}_i. \end{aligned} \quad (4.4)$$

Ya que los vectores de posición en la ecuación anterior se obtienen utilizando las ecuaciones (3.13), (3.15) y (3.16), estas se derivan en función del tiempo (respetando la regla de cadena y considerando que ϕ_i, a_i, l_i, p_i y v_i son valores **constantes**, por lo que sus derivadas en función del tiempo serán iguales a cero), lo que lleva a los siguientes resultados

$$\frac{d}{dt} \left(R_{\phi_i, z} \vec{n}_{a_i/0}^0 \right) = \frac{d}{dt} \begin{pmatrix} -\cos(\phi_i) \\ -\sin(\phi_i) \\ 0 \end{pmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \quad (4.5)$$

$$\begin{aligned} \dot{\vec{r}}_{i/a_i}^0 &= \begin{bmatrix} l_i \sin(\phi_i) \cdot \frac{d}{dt} [\sin(\theta_i)] \\ -l_i \cos(\phi_i) \cdot \frac{d}{dt} [\sin(\theta_i)] \\ -l_i \cdot \frac{d}{dt} [\cos(\theta_i)] \end{bmatrix} = \begin{bmatrix} l_i \sin(\phi_i) \cos(\theta_i) \dot{\theta}_i \\ -l_i \cos(\phi_i) \cos(\theta_i) \dot{\theta}_i \\ l_i \sin(\theta_i) \dot{\theta}_i \end{bmatrix} \\ &= \begin{bmatrix} l_i \sin(\phi_i) \cos(\theta_i) \\ -l_i \cos(\phi_i) \cos(\theta_i) \\ l_i \sin(\theta_i) \end{bmatrix} \dot{\theta}_i = \left[\left(R_{\phi_i, z} \vec{n}_{a_i/0}^0 \right) \times \vec{r}_{i/a_i}^0 \right] \dot{\theta}_i \end{aligned} \quad (4.6)$$

$$\begin{aligned} \dot{\vec{r}}_{p_i/i}^0 &= \omega_{p_i/i}^0 \times \vec{r}_{p_i/i}^0 = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} - \frac{d}{dt} \begin{pmatrix} -p_i \sin(\phi_i) \\ p_i \cos(\phi_i) \\ 0 \end{pmatrix} - \begin{pmatrix} a_i \sin(\phi_i) \\ -a_i \cos(\phi_i) \\ 0 \end{pmatrix} - \begin{bmatrix} l_i \sin(\phi_i) \cdot \frac{d}{dt} [\sin(\theta_i)] \\ -l_i \cos(\phi_i) \cdot \frac{d}{dt} [\sin(\theta_i)] \\ -l_i \cdot \frac{d}{dt} [\cos(\theta_i)] \end{bmatrix} \\ &= \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} - \begin{bmatrix} l_i \sin(\phi_i) \cos(\theta_i) \\ -l_i \cos(\phi_i) \cos(\theta_i) \\ l_i \sin(\theta_i) \end{bmatrix} \dot{\theta}_i = J^I(\vec{n}, \vec{r}) \dot{\theta} - \left[\left(R_{\phi_i, z} \vec{n}_{a_i/0}^0 \right) \times \vec{r}_{i/a_i}^0 \right] \dot{\theta}_i. \end{aligned} \quad (4.7)$$

Estos resultados, además de la ecuación (3.21), se sustituyen en la ecuación (4.4), de tal forma que se expande como sigue:

$$\begin{aligned} \vec{r}_{p_i/i}^T \dot{\vec{r}}_{e/0}^0 &= \left(R_{\phi_i, z} \vec{n}_{a_i/0}^0 \right)^T \left[\dot{\vec{r}}_{i/a_i}^0 \times \vec{r}_{p_i/i}^0 + \vec{r}_{i/a_i}^0 \times \dot{\vec{r}}_{p_i/i}^0 \right] \dot{\theta}_i - \vec{r}_{p_i/i}^T \dot{\vec{r}}_{e/0}^0 \\ &+ \left(R_{\phi_i, z} \vec{n}_{a_i/0}^0 \right)^T \left[\vec{r}_{i/a_i}^0 \times \vec{r}_{p_i/i}^0 \right] \ddot{\theta}_i, \end{aligned}$$

$$\begin{aligned} \vec{r}_{p_i/i}^T \dot{\vec{r}}_{e/0}^0 &= \left(R_{\phi_i, z} \vec{n}_{a_i/0}^0 \right)^T \left[\dot{\vec{r}}_{i/a_i}^0 \times \vec{r}_{p_i/i}^0 + \vec{r}_{i/a_i}^0 \times \dot{\vec{r}}_{p_i/i}^0 \right] \dot{\theta}_i \\ &+ \left(\left[\left(R_{\phi_i, z} \vec{n}_{a_i/0}^0 \right) \times \vec{r}_{i/a_i}^0 \right]^T \dot{\theta}_i - \dot{\theta}^T J^T \right) J \dot{\theta} + \left(R_{\phi_i, z} \vec{n}_{a_i/0}^0 \right)^T \left[\vec{r}_{i/a_i}^0 \times \vec{r}_{p_i/i}^0 \right] \ddot{\theta}_i. \end{aligned}$$

El resultado anterior contiene el término J , el cual es la misma matriz jacobiana geométrica definida en la ecuación (3.22). Asimismo, el producto entre dos vectores $\vec{g}^T \vec{h}$ será igual a $\vec{h}^T \vec{g}$, por lo tanto el resultado anterior se simplifica a

$$\begin{aligned} \vec{r}_{p_i/i}^T \ddot{\vec{r}}_{e/0}^0 &= \left(R_{\phi_i, z} \vec{n}_{a_i/0}^0 \right)^T \left[\dot{\vec{r}}_{i/a_i}^0 \times \vec{r}_{p_i/i}^0 + \vec{r}_{i/a_i}^0 \times \dot{\vec{r}}_{p_i/i}^0 \right] \dot{\theta}_i + \dot{\theta}^T J^T \left[\left(R_{\phi_i, z} \vec{n}_{a_i/0}^0 \right) \times \vec{r}_{i/a_i}^0 \right] \dot{\theta}_i \\ &\quad - \left(\dot{\theta}^T J^T \right) J \dot{\theta} + \left(R_{\phi_i, z} \vec{n}_{a_i/0}^0 \right)^T \left[\vec{r}_{i/a_i}^0 \times \dot{\vec{r}}_{p_i/i}^0 \right] \ddot{\theta}_i. \end{aligned}$$

Donde el término $\dot{\theta}^T J^T \left[\left(R_{\phi_i, z} \vec{n}_{a_i/0}^0 \right) \times \vec{r}_{i/a_i}^0 \right]$ se puede redefinir utilizando la regla del *triple producto escalar* [7], tal como se utilizó para la ecuación (3.14):

$$\begin{aligned} \vec{r}_{p_i/i}^T \ddot{\vec{r}}_{e/0}^0 &= \left(R_{\phi_i, z} \vec{n}_{a_i/0}^0 \right)^T \left[\dot{\vec{r}}_{i/a_i}^0 \times \vec{r}_{p_i/i}^0 + \vec{r}_{i/a_i}^0 \times \dot{\vec{r}}_{p_i/i}^0 \right] \dot{\theta}_i + \left(R_{\phi_i, z} \vec{n}_{a_i/0}^0 \right)^T \left[\vec{r}_{i/a_i}^0 \times J \dot{\theta} \right] \dot{\theta}_i \\ &\quad - \left(\dot{\theta}^T J^T \right) J \dot{\theta} + \left(R_{\phi_i, z} \vec{n}_{a_i/0}^0 \right)^T \left[\vec{r}_{i/a_i}^0 \times \dot{\vec{r}}_{p_i/i}^0 \right] \ddot{\theta}_i, \end{aligned}$$

$$\begin{aligned} \vec{r}_{p_i/i}^T \ddot{\vec{r}}_{e/0}^0 &= \left(R_{\phi_i, z} \vec{n}_{a_i/0}^0 \right)^T \left[\dot{\vec{r}}_{i/a_i}^0 \times \vec{r}_{p_i/i}^0 + \vec{r}_{i/a_i}^0 \times \left(\dot{\vec{r}}_{p_i/i}^0 + J \dot{\theta} \right) \right] \dot{\theta}_i \\ &\quad - \left(\dot{\theta}^T J^T \right) J \dot{\theta} + \left(R_{\phi_i, z} \vec{n}_{a_i/0}^0 \right)^T \left[\vec{r}_{i/a_i}^0 \times \dot{\vec{r}}_{p_i/i}^0 \right] \ddot{\theta}_i. \quad (4.8) \end{aligned}$$

Este último resultado puede expandirse aún más si se sustituyen las ecuaciones (4.6) y (4.7), sin embargo, eso llevaría a un resultado bastante extenso. Lo que sigue es definir este resultado para cada una de las articulaciones en el robot, es decir

$$\begin{aligned} \vec{r}_{p_1/1}^T \ddot{\vec{r}}_{e/0}^0 &= \left(R_{0, z} \vec{n}_{a_1/0}^0 \right)^T \left[\dot{\vec{r}}_{1/a_1}^0 \times \vec{r}_{p_1/1}^0 + \vec{r}_{1/a_1}^0 \times \left(\dot{\vec{r}}_{p_1/1}^0 + J \dot{\theta} \right) \right] \dot{\theta}_1 \\ &\quad - \left(\dot{\theta}^T J^T \right) J \dot{\theta} + \left(R_{0, z} \vec{n}_{a_1/0}^0 \right)^T \left[\vec{r}_{1/a_1}^0 \times \dot{\vec{r}}_{p_1/1}^0 \right] \ddot{\theta}_1, \quad (4.9) \end{aligned}$$

$$\begin{aligned} \vec{r}_{p_2/2}^T \ddot{\vec{r}}_{e/0}^0 &= \left(R_{\frac{2}{3}\pi, z} \vec{n}_{a_2/0}^0 \right)^T \left[\dot{\vec{r}}_{2/a_2}^0 \times \vec{r}_{p_2/2}^0 + \vec{r}_{2/a_2}^0 \times \left(\dot{\vec{r}}_{p_2/2}^0 + J \dot{\theta} \right) \right] \dot{\theta}_2 \\ &\quad - \left(\dot{\theta}^T J^T \right) J \dot{\theta} + \left(R_{\frac{2}{3}\pi, z} \vec{n}_{a_2/0}^0 \right)^T \left[\vec{r}_{2/a_2}^0 \times \dot{\vec{r}}_{p_2/2}^0 \right] \ddot{\theta}_2, \quad (4.10) \end{aligned}$$

$$\begin{aligned} \vec{r}_{p_3/3}^T \ddot{\vec{r}}_{e/0}^0 &= \left(R_{\frac{4}{3}\pi, z} \vec{n}_{a_3/0}^0 \right)^T \left[\dot{\vec{r}}_{3/a_3}^0 \times \vec{r}_{p_3/3}^0 + \vec{r}_{3/a_3}^0 \times \left(\dot{\vec{r}}_{p_3/3}^0 + J \dot{\theta} \right) \right] \dot{\theta}_3 \\ &\quad - \left(\dot{\theta}^T J^T \right) J \dot{\theta} + \left(R_{\frac{4}{3}\pi, z} \vec{n}_{a_3/0}^0 \right)^T \left[\vec{r}_{3/a_3}^0 \times \dot{\vec{r}}_{p_3/3}^0 \right] \ddot{\theta}_3, \quad (4.11) \end{aligned}$$

que al agrupar las ecuaciones anteriores, se obtiene

$$\underbrace{\begin{bmatrix} \vec{r}_{p_1/1}^0 & \vec{r}_{p_2/2}^0 & \vec{r}_{p_3/3}^0 \end{bmatrix}^T}_{J, \text{ ecuación (3.23)}} \ddot{\vec{r}}_{e/0}^0 = \begin{bmatrix} \left(R_{0, z} \vec{n}_{a_1/0}^0 \right)^T \left[\dot{\vec{r}}_{1/a_1}^0 \times \vec{r}_{p_1/1}^0 + \vec{r}_{1/a_1}^0 \times \left(\dot{\vec{r}}_{p_1/1}^0 + J \dot{\theta} \right) \right] & 0 & 0 \\ 0 & \left(R_{\frac{2}{3}\pi, z} \vec{n}_{a_2/0}^0 \right)^T \left[\dot{\vec{r}}_{2/a_2}^0 \times \vec{r}_{p_2/2}^0 + \vec{r}_{2/a_2}^0 \times \left(\dot{\vec{r}}_{p_2/2}^0 + J \dot{\theta} \right) \right] & 0 \\ 0 & 0 & \left(R_{\frac{4}{3}\pi, z} \vec{n}_{a_3/0}^0 \right)^T \left[\dot{\vec{r}}_{3/a_3}^0 \times \vec{r}_{p_3/3}^0 + \vec{r}_{3/a_3}^0 \times \left(\dot{\vec{r}}_{p_3/3}^0 + J \dot{\theta} \right) \right] \end{bmatrix} \dot{\theta} - \begin{bmatrix} \left(\dot{\theta}^T J^T \right) J \\ \left(\dot{\theta}^T J^T \right) J \\ \left(\dot{\theta}^T J^T \right) J \end{bmatrix} \dot{\theta} + \underbrace{\begin{bmatrix} \left(R_{0, z} \vec{n}_{a_1/0}^0 \right)^T \left[\vec{r}_{1/a_1}^0 \times \dot{\vec{r}}_{p_1/1}^0 \right] & 0 & 0 \\ 0 & \left(R_{\frac{2}{3}\pi, z} \vec{n}_{a_2/0}^0 \right)^T \left[\vec{r}_{2/a_2}^0 \times \dot{\vec{r}}_{p_2/2}^0 \right] & 0 \\ 0 & 0 & \left(R_{\frac{4}{3}\pi, z} \vec{n}_{a_3/0}^0 \right)^T \left[\vec{r}_{3/a_3}^0 \times \dot{\vec{r}}_{p_3/3}^0 \right] \end{bmatrix}}_{J, \text{ ecuación (3.24)}} \ddot{\theta}. \quad (4.12)$$

Si los términos a la izquierda y derecha de la ecuación anterior se agrupan, se puede reducir el resultado a la ecuación (4.2), donde $J^I(\vec{n}, \vec{r})$ representa a la derivada de la matriz jacobiana **geométrica**, la cual está definida de la siguiente manera

$$j^I(\vec{n}, \vec{r}) = J_{\vec{r}}^{-1} \left(\begin{bmatrix} (R_{0,z} \vec{n}_{a_i/0}^0)^T [\dot{r}_{i/a_i}^0 \times \vec{r}_{p_i/i}^0 + \vec{r}_{i/a_i}^0 \times (\dot{r}_{p_i/i}^0 + J\dot{\theta})] & 0 & 0 \\ 0 & (R_{\frac{1}{2}\pi, z} \vec{n}_{a_2/0}^0)^T [\dot{r}_{2/a_2}^0 \times \vec{r}_{p_2/2}^0 + \vec{r}_{2/a_2}^0 \times (\dot{r}_{p_2/2}^0 + J\dot{\theta})] & 0 \\ 0 & 0 & (R_{\frac{3}{4}\pi, z} \vec{n}_{a_3/0}^0)^T [\dot{r}_{3/a_3}^0 \times \vec{r}_{p_3/3}^0 + \vec{r}_{3/a_3}^0 \times (\dot{r}_{p_3/3}^0 + J\dot{\theta})] \end{bmatrix} - \begin{bmatrix} (\dot{\theta}^T J^T) J \\ (\dot{\theta}^T J^T) J \\ (\dot{\theta}^T J^T) J \end{bmatrix} \right) \quad (4.13)$$

siendo $J_{\vec{r}}$ la sub-matriz jacobiana que define los efectos lineales en el extremo del robot. Cada término dentro de la matriz se calcula directamente con las ecuaciones (3.15), (3.16), (3.22), (4.6) y (4.7). Su código en MATLAB, ejemplificado en el algoritmo 4, puede consultarse en el apéndice A.4.

Algoritmo 4: Derivada de la Matriz Jacobiana Geométrica

```

Entrada:  $(\vec{r}_{e/0}^0, \bar{\theta}, \dot{\theta}, a, l, p)$  ; /* Parámetros geométricos */
 $\phi \leftarrow \frac{2}{3}\pi$  ; /* Separación entre articulaciones */
 $J^I(\vec{n}, \vec{r}) \leftarrow$  algoritmo 2
 $J_{\vec{r}} \leftarrow$  zeros(3, 3)
 $J_{\bar{\theta}} \leftarrow$  zeros(3, 3) ; /* Inicialización de las matrices jacobianas */
for  $i \leftarrow 1$  to 3 do
     $R_{\phi_i, z} \vec{n}_{a_i/0}^0 \leftarrow \begin{bmatrix} -\cos(\phi_i) \\ -\sin(\phi_i) \\ 0 \end{bmatrix}$  ; /* Eje de acción de la  $i$ -ésima articulación */
     $\vec{r}_{i/a_i}^0 \leftarrow$  ecuación (3.15)
     $\dot{r}_{i/a_i}^0 \leftarrow$  ecuación (4.6)
     $\vec{r}_{p_i/i}^0 \leftarrow$  ecuación (3.16)
     $\dot{r}_{p_i/i}^0 \leftarrow$  ecuación (4.7)
     $J_{\vec{r}}(i, :) \leftarrow \vec{r}_{p_i/i}^{0T}$ 
     $J_{\bar{\theta}}(i, i) \leftarrow (R_{\phi_i, z} \vec{n}_{a_i/0}^0)^T [\dot{r}_{i/a_i}^0 \times \vec{r}_{p_i/i}^0 + \vec{r}_{i/a_i}^0 \times (\dot{r}_{p_i/i}^0 + J\dot{\theta})]$ 
end
 $J^I(\vec{n}, \vec{r}) \leftarrow J_{\vec{r}}^{-1} \left( J_{\bar{\theta}} - [(\dot{\theta}^T J^T) J \quad (\dot{\theta}^T J^T) J \quad (\dot{\theta}^T J^T) J]^T \right)$  ; /* ecuación (4.13) */
Salida:  $J(\vec{n}, \vec{r})$ 

```

4.2. Velocidad Instantánea de las Articulaciones

Los resultados del cálculo de la cinemática inversa (figura 3.2) pueden ser utilizados para calcular la velocidad instantánea de las articulaciones; para ello, solo se derivan los valores numéricos utilizando la siguiente ecuación [12]

$$\dot{\bar{\theta}}(t) = \frac{\bar{\theta}(t + \Delta t) - \bar{\theta}(t)}{\Delta t}, \quad (4.14)$$

con $t \in \mathbb{R}^+$, que representa el tiempo de simulación del sistema. Esto se representa en el algoritmo 5 y su código en MATLAB se muestra en el apéndice A.5.

Algoritmo 5: Velocidad instantánea las Articulaciones

```
Entrada:  $(\bar{\theta}, t)$  ; /* Posición de las Articulaciones y Tiempo de Simulación */  
[m, n]  $\leftarrow$  size  $(\bar{\theta})$  ; /* Obtiene el número de elementos calculados */  
dt  $\leftarrow$   $\frac{t}{n}$  ; /* Calcula el tamaño de paso para la derivada numérica */  
 $\dot{\theta} \leftarrow$  zeros(m, n) ; /* Inicializa la velocidad con ceros */  
for j  $\leftarrow$  1 to n do  
 |  $\dot{\theta}(:, j) \leftarrow \frac{\bar{\theta}(:, j+1) - \bar{\theta}(:, j)}{\Delta t}$  ; /* Calcula la ecuación (4.14) */  
end  
Salida:  $\dot{\theta}(t)$ 
```

Para ejemplificar su uso, se utilizan los resultados de la figura 3.2, además de considerar que la trayectoria de las articulaciones se ejecuta en un tiempo $t = 10$ segundos, por lo tanto, el resultado del cálculo se representa, gráficamente, como se muestra a continuación.

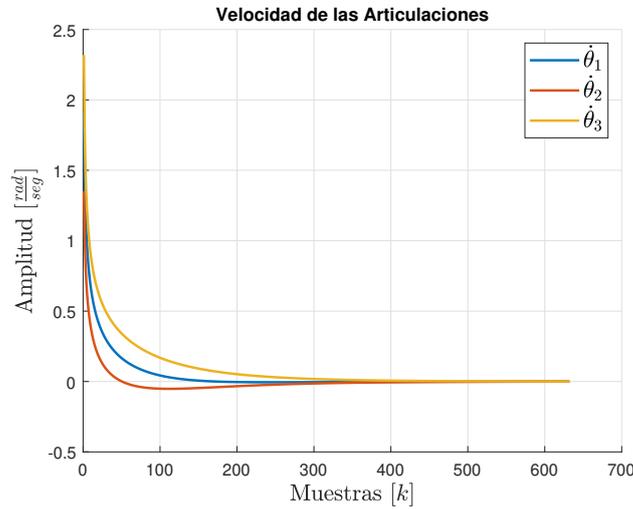


Figura 4.1: Velocidad Instantánea las Articulaciones

4.3. Velocidad Instantánea del Efector Final

Utilizando la velocidad instantánea de las articulaciones (figura 4.1) y la matriz jacobiana (ecuación (3.22) y algoritmo 2), la velocidad instantánea del efector final puede ser calculada con la ecuación (4.1), lo cual se representa con el algoritmo 6 y su código en MATLAB (apéndice A.6).

Algoritmo 6: Velocidad Instantánea del efector final

```
Entrada:  $(\bar{\theta}, \dot{\bar{\theta}}, a, l, p)$  ; /* Parámetros geométricos */
 $[\sim, n] \leftarrow \text{size}(\bar{\theta})$  ; /* Obtiene el número de elementos calculados */
 $\dot{\bar{r}}_{e/0}^0(t) \leftarrow \text{zeros}(m, n)$  ; /* Inicializa la velocidad con ceros */
for  $j \leftarrow 1$  to  $n$  do
     $\dot{\bar{r}}_{e/0}^0(t) \leftarrow$  algoritmo 1 ; /* Cinemática Directa */
     $J^I(\bar{n}, \bar{r}) \leftarrow$  algoritmo 2 ; /* Calcula la matriz jacobiana geométrica */
     $\dot{\bar{r}}_{e/0}^0(:, j) \leftarrow$  ecuación (4.1) ; /* Calcula la velocidad */
end
Salida:  $\dot{\bar{r}}_{e/0}^0(t)$ 
```

Para ejemplificar su uso, se utilizan los resultados de la figura 4.1, por lo tanto, el resultado del cálculo se representa, gráficamente, como se muestra a continuación

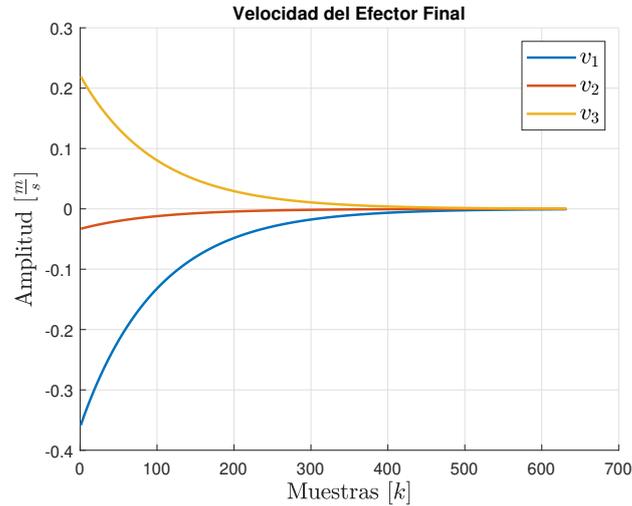


Figura 4.2: Velocidad Instantánea del efector final

4.4. Aceleración Instantánea de las Articulaciones

Los resultados del cálculo de la velocidad instantánea de las articulaciones (figura 4.1) pueden ser utilizados para calcular la aceleración instantánea de las mismas; para ello, solo se derivan los valores numéricos utilizando la siguiente ecuación [12]

$$\ddot{\bar{\theta}}(t) = \frac{\dot{\bar{\theta}}(t + \Delta t) - \dot{\bar{\theta}}(t)}{\Delta t}, \quad (4.15)$$

con $t \in \mathbb{R}^+$, que representa el tiempo de simulación del sistema. Esto se representa en el algoritmo 7 y su código en MATLAB se muestra en el apéndice A.7.

Algoritmo 7: Aceleración Instantánea de las Articulaciones

```
Entrada:  $(\dot{\theta}, t)$  ; /* Velocidad de las Articulaciones y Tiempo de Simulación */  
[m, n]  $\leftarrow$  size ( $\theta$ ) ; /* Obtiene el número de elementos calculados */  
dt  $\leftarrow$   $\frac{t}{n}$  ; /* Calcula el tamaño de paso para la derivada numérica */  
 $\ddot{\theta} \leftarrow$  zeros(m, n) ; /* Inicializa la aceleración con ceros */  
Itera sobre el  $i$ -ésimo valor de  $\bar{\theta}$   
for  $j \leftarrow 1$  to  $n$  do  
|  $\ddot{\theta}(:, j) \leftarrow \frac{\dot{\theta}(:, j+1) - \dot{\theta}(:, j)}{\Delta t}$  ; /* Calcula la ecuación (4.15) */  
end  
Salida:  $\ddot{\theta}(t)$ 
```

Para ejemplificar su uso, se utilizan los resultados de la figura 4.1, además de considerar que la trayectoria de las articulaciones se ejecuta en un tiempo $t = 10$ segundos, por lo tanto, el resultado del cálculo se representa, gráficamente, como se muestra a continuación

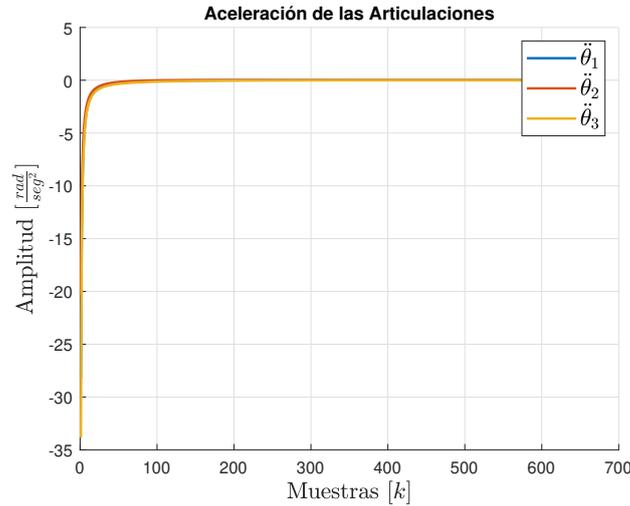


Figura 4.3: Aceleración Instantánea de las Articulaciones

4.5. Aceleración Instantánea del Efector Final

Utilizando la aceleración instantánea de las articulaciones (figura 4.3), la matriz jacobiana geométrica (ecuación (3.22) y algoritmo 2) y su respectiva derivada (ecuación (4.13) y algoritmo 4), la aceleración instantánea del efector final puede ser calculada con la ecuación (4.2), lo cual se representa con el algoritmo 8 y su código en MATLAB (apéndice A.8). Para ejemplificar su uso, se utilizan los resultados de la figura 4.1, por lo tanto, el resultado del cálculo se representa, gráficamente, como se muestra en la figura 4.4.

Algoritmo 8: Aceleración instantánea del Efecto Final

```
Entrada:  $(\bar{\theta}, \dot{\bar{\theta}}, \ddot{\bar{\theta}}, a, l, p, v, \Delta\bar{\theta})$  ; /* Parámetros geométricos */  
[~, n]  $\leftarrow$  size ( $\bar{\theta}$ ) ; /* Obtiene el número de elementos calculados */  
 $\ddot{r}_{e/0}^0(t) \leftarrow$  zeros(m, n) ; /* Inicializa la aceleración con ceros */  
for  $j \leftarrow 1$  to  $n$  do  
     $\ddot{r}_{e/0}^0(t) \leftarrow$  algoritmo 1 ; /* Cinemática Directa */  
     $J^I(\bar{n}, \bar{r}) \leftarrow$  algoritmo 2 ; /* Calcula la matriz jacobiana geométrica */  
     $\dot{J}^I(\bar{n}, \bar{r}) \leftarrow$  algoritmo 4 ; /* Calcula la derivada de la matriz jacobiana */  
     $\ddot{r}_{e/0}^0(:, j) \leftarrow$  ecuación (4.2) ; /* Calcula la aceleración instantánea */  
end  
Salida:  $\ddot{r}_{e/0}^0(t)$ 
```

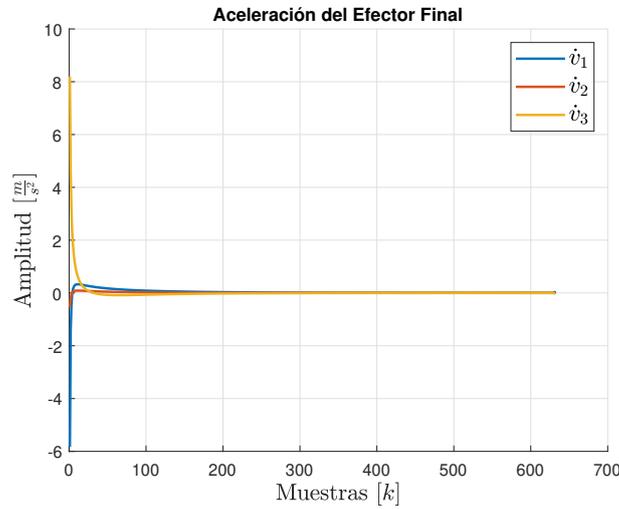


Figura 4.4: Aceleración Instantánea del Efecto Final

5. Análisis Dinámico

En robots con múltiples cuerpos rígidos acoplados, las fuerzas y torques se propagan y concentran en cada uno de los centros de masa [11], por lo que es común estudiar el comportamiento de un robot serial en cada uno de los centros de masa de sus eslabones, sin embargo, esto se vuelve una tarea demasiado compleja en un robot paralelo [13], por lo que se propone calcular las posiciones y velocidades en cada uno de los puntos de unión entre los cuerpos rígidos

5.1. Posiciones y Velocidades en el Robot

Tal como se menciona en [14], los Robots Paralelos tipo Delta constan de tres tipos de cuerpos rígidos: l_i , v_i y p_i (figura 1.1). Comenzando por el efecto final, este tiene a su centro de masa exactamente en su punto central, ya que toda su geometría está distribuida uniformemente respecto a este (figura 5.1).

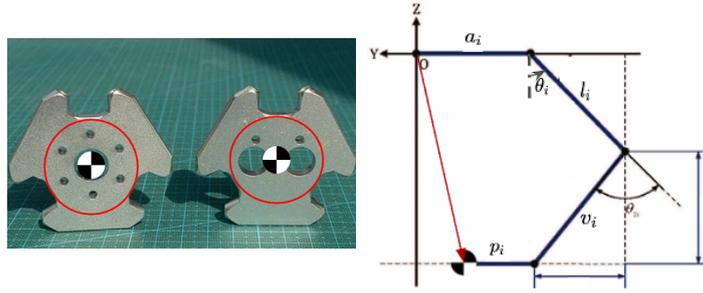


Figura 5.1: Centro de Masa del Efecto Final.

Además, ya que la cinemática directa y la matriz jacobiana geométrica $J^I(\vec{n}, \vec{r})$ se calculan hasta el centro de este cuerpo rígido, se puede concluir que su posición y velocidad $\vec{r}_{e/0}^0, \dot{\vec{r}}_{e/0}^0 \in \mathbb{R}^{3 \times 1}$ serán iguales a las calculadas con las ecuaciones (2.35) a (2.37) y (3.21), es decir

$$\vec{r}_{e/0}^0 = \vec{r}_{\bullet e/0}^0 \quad (5.1)$$

$$\dot{\vec{r}}_{e/0}^0 = \dot{\vec{r}}_{\bullet e/0}^0 = J^I(\vec{n}, \vec{r}) \dot{\theta}. \quad (5.2)$$

Asimismo, debido a que el movimiento del efecto final es puramente lineal, su velocidad angular es igual a cero. El siguiente punto de análisis es aquél donde el motor y la base del robot se conectan con el eslabón l_i .

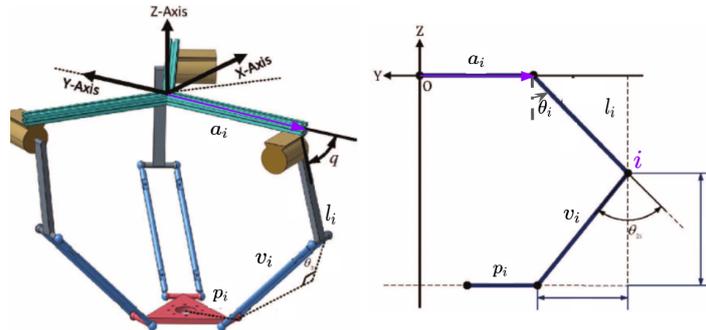


Figura 5.2: Punto de Unión entre la Base del Robot y el Cuerpo Rígido l_i

Si, y solo si la base del robot se desplazara, la velocidad lineal del punto señalado en la figura anterior tendría un valor numérico, de otra forma, la velocidad lineal será igual a cero, sin embargo, existe una velocidad angular $\omega_{m_i/0}^0 \in \mathbb{R}^{3 \times 1}$ ocasionada por el giro del motor, esta se describe como

$$\omega_{m_i/0}^0 = (R_{\phi_i, z} \vec{n}_{a_i/0}^0) \dot{\theta}_i = \begin{bmatrix} -\cos(\phi_i) \\ -\sin(\phi_i) \\ 0 \end{bmatrix} \dot{\theta}_i, \quad (5.3)$$

donde $R_{\phi_i, z}$ y $\vec{n}_{a_i/0}^0$ se definieron en la ecuaciones (2.6) y (3.12), mientras que $\dot{\theta}_i \in \mathbb{R}$ representa a la velocidad de la i -ésima articulación. Debido a que la velocidad en el efecto final se define utilizando

una matriz jacobiana, esto también se puede considerar para todos los motores en el sistema, donde ϕ_i será igual a $0, \frac{2}{3}\pi$ y $\frac{4}{3}\pi$ radianes cuando $i = 1, 2, 3$. Por lo tanto, sus velocidades angulares se definen de la siguiente manera

$$\begin{aligned}\omega_{m_1/0}^0 &= \left[\begin{matrix} (R_{0,z} \vec{n}_{a_1/0}^0) & 0_{3 \times 1} & 0_{3 \times 1} \end{matrix} \right] \dot{\theta} = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \dot{\theta} \\ &= J_{\omega_{m_1}}^I(\vec{n}, \vec{r}) \dot{\theta}\end{aligned}\quad (5.4)$$

$$\begin{aligned}\omega_{m_2/0}^0 &= \left[0_{3 \times 1} \quad \left(R_{\frac{2}{3}\pi, z} \vec{n}_{a_2/0}^0 \right) \quad 0_{3 \times 1} \right] \dot{\theta} = \begin{bmatrix} 0 & \frac{1}{2} & 0 \\ 0 & -\frac{\sqrt{3}}{2} & 0 \\ 0 & 0 & 0 \end{bmatrix} \dot{\theta} \\ &= J_{\omega_{m_2}}^I(\vec{n}, \vec{r}) \dot{\theta}\end{aligned}\quad (5.5)$$

$$\begin{aligned}\omega_{m_3/0}^0 &= \left[0_{3 \times 1} \quad 0_{3 \times 1} \quad \left(R_{\frac{4}{3}\pi, z} \vec{n}_{a_3/0}^0 \right) \right] \dot{\theta} = \begin{bmatrix} 0 & 0 & \frac{1}{2} \\ 0 & 0 & \frac{\sqrt{3}}{2} \\ 0 & 0 & 0 \end{bmatrix} \dot{\theta} \\ &= J_{\omega_{m_3}}^I(\vec{n}, \vec{r}) \dot{\theta}\end{aligned}\quad (5.6)$$

Con estas velocidades angulares se puede calcular la velocidad lineal en el extremo del cuerpo rígido l_i :

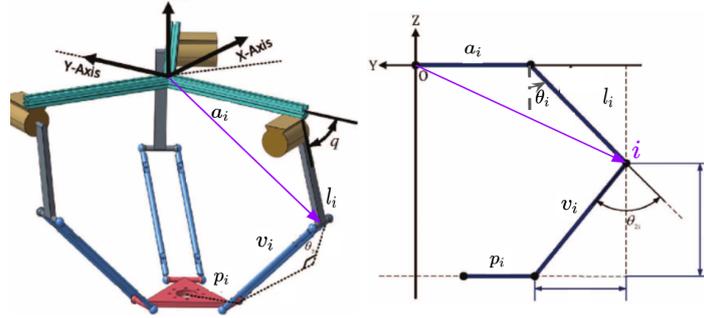


Figura 5.3: Punto Extremo del Cuerpo Rígido l_i

Su postura (posición y orientación) se definió en secciones anteriores con la ecuación (2.7), donde la posición del punto de análisis se estableció en la ecuación (2.8). Para conocer su velocidad, se deriva la ecuación mencionada en función del tiempo, lo que lleva al siguiente resultado

$$\dot{\vec{r}}_{i/0}^0 = \omega_{m_i/0}^0 \times \vec{r}_{i/a_i}^0 = \left[\left(R_{\phi_i, z} \vec{n}_{a_i/0}^0 \right) \times \vec{r}_{i/a_i}^0 \right] \dot{\theta}_i = \begin{bmatrix} l_i \sin(\phi_i) \cos(\theta_i) \\ -l_i \cos(\phi_i) \cos(\theta_i) \\ l_i \sin(\theta_i) \end{bmatrix} \dot{\theta}_i \quad (5.7)$$

donde $\vec{r}_{i/a_i}^0 \in \mathbb{R}^{3 \times 1}$ se definió en la ecuación (3.15). Debido a que la velocidad en el efector final se define utilizando una matriz jacobiana, esto también se puede considerar para cada cuerpo rígido l_i ,

donde ϕ_i será igual a $0, \frac{2}{3}\pi$ y $\frac{4}{3}\pi$ radianes cuando $i = 1, 2, 3$. Por lo tanto, sus velocidades son

$$\begin{aligned}\dot{\vec{r}}_{1/0}^0 &= \left[\left(R_{0,z} \vec{n}_{a_1/0}^0 \right) \times \vec{r}_{1/a_1}^0 \quad 0_{3 \times 1} \quad 0_{3 \times 1} \right] \dot{\theta} = \begin{bmatrix} 0 & 0 & 0 \\ -l_1 \cos(\theta_1) & 0 & 0 \\ l_1 \sin(\theta_1) & 0 & 0 \end{bmatrix} \dot{\theta} \\ &= J_{v_1}^I(\vec{n}, \vec{r}) \dot{\theta}\end{aligned}\quad (5.8)$$

$$\begin{aligned}\dot{\vec{r}}_{2/0}^0 &= \left[0_{3 \times 1} \quad \left(R_{\frac{2}{3}\pi, z} \vec{n}_{a_2/0}^0 \right) \times \vec{r}_{2/a_2}^0 \quad 0_{3 \times 1} \right] \dot{\theta} = \begin{bmatrix} 0 & \frac{\sqrt{3}}{2} l_2 \cos(\theta_2) & 0 \\ 0 & \frac{1}{2} l_2 \cos(\theta_2) & 0 \\ 0 & l_2 \sin(\theta_2) & 0 \end{bmatrix} \dot{\theta} \\ &= J_{v_2}^I(\vec{n}, \vec{r}) \dot{\theta}\end{aligned}\quad (5.9)$$

$$\begin{aligned}\dot{\vec{r}}_{3/0}^0 &= \left[0_{3 \times 1} \quad 0_{3 \times 1} \quad \left(R_{\frac{4}{3}\pi, z} \vec{n}_{a_3/0}^0 \right) \times \vec{r}_{3/a_3}^0 \right] \dot{\theta} = \begin{bmatrix} 0 & 0 & -\frac{\sqrt{3}}{2} l_3 \cos(\theta_3) \\ 0 & 0 & \frac{1}{2} l_3 \cos(\theta_3) \\ 0 & 0 & l_3 \sin(\theta_3) \end{bmatrix} \dot{\theta} \\ &= J_{v_3}^I(\vec{n}, \vec{r}) \dot{\theta}\end{aligned}\quad (5.10)$$

Debido a que existen velocidades lineales, generadas por el movimiento rotacional del motor, la velocidad angular del cuerpo l_i será la misma que la del motor m_i , esto es

$$\omega_{i/0}^0 = \omega_{m_i/0}^0 = \left(R_{\phi_i, z} \vec{n}_{a_i/0}^0 \right) \dot{\theta}_i = \begin{bmatrix} -\cos(\phi_i) \\ -\sin(\phi_i) \\ 0 \end{bmatrix} \dot{\theta}_i, \quad (5.11)$$

que para cada cuerpo rígido, sus respectivas matrices jacobianas son las mismas que las de las ecuaciones (5.4) a (5.6) Para el análisis dinámico, lo más importante es conocer las matrices jacobianas, por esa razón, en el algoritmo 9 y el apéndice A.9 se muestra cómo calcular estos términos con MATLAB.

Algoritmo 9: Matrices Jacobianas del Cuerpo Rígido l_i

```

Entrada:  $(\bar{\theta}, l_i)$  ; /* Parámetros geométricos */
 $\phi \leftarrow \frac{2}{3}\pi$  ; /* Separación entre articulaciones */
 $J_v \leftarrow \text{cell}(3, 1)$ 
 $J_\omega \leftarrow \text{cell}(3, 1)$  ; /* Inicializa celdas para almacenar las matrices */
for  $i \leftarrow 1$  to 3 do
   $J_{v_i} \leftarrow \text{zeros}(3, 3)$ 
   $J_{\omega_i} \leftarrow \text{zeros}(3, 3)$  ; /* Inicialización de las matrices jacobianas */
   $R_{\phi_i, z} \vec{n}_{a_i/0}^0 \leftarrow \begin{bmatrix} -\cos(\phi_i) \\ -\sin(\phi_i) \\ 0 \end{bmatrix}$  ; /* Eje de acción de la  $i$ -ésima articulación */
   $\vec{r}_{i/a_i}^0 \leftarrow$  ecuación (3.15)
   $J_{v_i}(:, i) \leftarrow (R_{\phi_i, z} \vec{n}_{a_i/0}^0) \times \vec{r}_{i/a_i}^0$  ; /* Matriz jacobiana de velocidad lineal */
   $J_v\{i\} \leftarrow J_{v_i}$  ; /* Almacena el valor en la  $i$ -ésima celda */
   $J_{\omega_i}(:, i) \leftarrow R_{\phi_i, z} \vec{n}_{a_i/0}^0$  ; /* Matriz jacobiana de velocidad angular */
   $J_\omega\{i\} \leftarrow J_{\omega_i}$  ; /* Almacena el valor en la  $i$ -ésima celda */
end
Salida:  $(J_v, J_\omega)$ 

```

El último punto por analizar es la conexión entre el efector final y el eslabón virtual (figura 5.4), que comparten la misma velocidad lineal, esto es

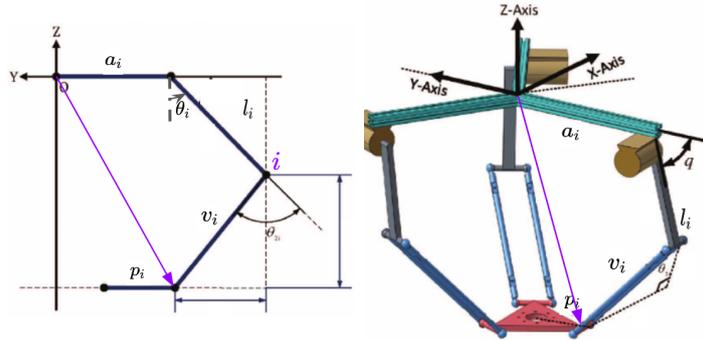


Figura 5.4: Punto Extremo del Cuerpo Rígido v_i

$$\dot{\vec{r}}_{p_i/0}^0 = \dot{\vec{r}}_{e/0}^0 = J^I(\vec{n}, \vec{r}) \dot{\theta}, \quad (5.12)$$

donde el vector de posición $\vec{r}_{p_i/0}^0$ se calcula de manera similar a $\vec{r}_{p_i/i}^0$ en la ecuación (3.16):

$$\begin{aligned}
\vec{r}_{p_i/0}^0 &= \vec{r}_{e/0}^0 - R_{\phi_i,z} T_{p_i,y} \\
&= \begin{bmatrix} x \\ y \\ z \end{bmatrix} - \begin{bmatrix} -p_i \sin(\phi_i) \\ p_i \cos(\phi_i) \\ 0 \end{bmatrix} \\
&= \begin{bmatrix} x + p_i \sin(\phi_i) \\ y - p_i \cos(\phi_i) \\ z \end{bmatrix}.
\end{aligned} \tag{5.13}$$

Vale la pena recordar que **el efector final no gira, solo se desplaza linealmente**, por lo que se considera que la velocidad angular en el punto de análisis es igual a cero. **Esto no significa que el eslabón virtual no gira, sino que el punto de análisis tiene una velocidad angular igual a cero**; si se estudiara el centro de masa del eslabón, se encontraría una velocidad angular que podría utilizarse en el análisis de las energías cinética y potencial.

5.2. Energía Cinética y Potencial

Para estudiar el modelo de casi cualquier sistema dinámico, se utilizan las energías cinética y potencial. La primera surge a partir del movimiento de uno o más cuerpos rígidos, mientras que la segunda depende de la posición de estos respecto a un marco de referencia inercial, afectado por una aceleración gravitacional; en ambos casos, los valores de la energía son escalares y **en el estudio de sistemas con múltiples cuerpos rígidos, las energías se analizan en sus centros de masa, pero como se explicó en la sección anterior, el análisis en los centros de masa, para este tipo de robots, se vuelve más complicado**, por lo que se propone estudiar las energías en los mismos puntos donde se calcularon las velocidades en la sección anterior. Estas energías comúnmente se describen como

$$K_{j/0}^0 = \frac{1}{2} m_j \dot{\vec{r}}_{j/0}^T \dot{\vec{r}}_{j/0}^0 + \frac{1}{2} \omega_{j/0}^T \mathbf{I}_{j/0}^0 \omega_{j/0}^0 \tag{5.14}$$

$$P_{j/0}^0 = m_j g^T \vec{r}_{j/0}^0, \tag{5.15}$$

donde $m_j \in \mathbb{R}$ es la masa del cuerpo rígido estudiado, mientras que $g \in \mathbb{R}^{3 \times 1}$ representa el vector de aceleración gravitacional que afecta al sistema estudiado; además, $\dot{\vec{r}}_{j/0}^0, \omega_{j/0}^0 \in \mathbb{R}^{3 \times 1}$ son la velocidad lineal y angular del j -ésimo punto estudiado dentro del robot; por otro lado, $\mathbf{I}_{j/0}^0 \in \mathbb{R}^{3 \times 3}$ representa al **tensor de inercia** del cuerpo estudiado respecto al punto de análisis de robot, expresado en coordenadas inerciales. Para el sistema que se está estudiando, la energía cinética requiere de las

ecuaciones (3.21), (5.4) a (5.6) y (5.8) a (5.10), esta se expresa y reduce de la siguiente manera

$$\begin{aligned}
K_{j/0}^0 &= \underbrace{\frac{1}{2}\omega_{m_1/0}^T \mathbf{I}_{m_1/0}^0 \omega_{m_1/0}^0 + \frac{1}{2}\omega_{m_2/0}^T \mathbf{I}_{m_2/0}^0 \omega_{m_2/0}^0 + \frac{1}{2}\omega_{m_3/0}^T \mathbf{I}_{m_3/0}^0 \omega_{m_3/0}^0}_{\text{energías rotacionales al inicio de } l_i, \text{ velocidades (5.4) a (5.6)}} \\
&+ \underbrace{\frac{1}{2}m_{l_1} \dot{r}_{1/0}^T \dot{r}_{1/0}^0 + \frac{1}{2}m_{l_2} \dot{r}_{2/0}^T \dot{r}_{2/0}^0 + \frac{1}{2}m_{l_3} \dot{r}_{3/0}^T \dot{r}_{3/0}^0 + \frac{1}{2}\omega_{l_1/0}^T \mathbf{I}_{l_1/0}^0 \omega_{l_1/0}^0 + \frac{1}{2}\omega_{l_2/0}^T \mathbf{I}_{l_2/0}^0 \omega_{l_2/0}^0 + \frac{1}{2}\omega_{l_3/0}^T \mathbf{I}_{l_3/0}^0 \omega_{l_3/0}^0}_{\text{energías lineales y rotacionales al extremo de } l_i, \text{ velocidades (5.4) a (5.6) y (5.8) a (5.10)}} \\
&+ \underbrace{\frac{1}{2}m_{v_1} \dot{r}_{1/0}^T \dot{r}_{1/0}^0 + \frac{1}{2}m_{v_2} \dot{r}_{2/0}^T \dot{r}_{2/0}^0 + \frac{1}{2}m_{v_3} \dot{r}_{3/0}^T \dot{r}_{3/0}^0}_{\text{energías lineales al inicio de } v_i, \text{ velocidades (5.8) a (5.10)}} + \underbrace{\frac{1}{2}m_{v_1} \dot{r}_{p_1/0}^T \dot{r}_{p_1/0}^0 + \frac{1}{2}m_{v_2} \dot{r}_{p_2/0}^T \dot{r}_{p_2/0}^0 + \frac{1}{2}m_{v_3} \dot{r}_{p_3/0}^T \dot{r}_{p_3/0}^0}_{\text{energías lineales al extremo de } v_i, \text{ velocidad (3.21)}} \\
&+ \underbrace{\frac{1}{2}m_e \dot{r}_{e/0}^T \dot{r}_{e/0}^0}_{\text{energía del efector final, velocidad (3.21)}}
\end{aligned}$$

estos resultados podrían expandirse sustituyendo las matrices jacobianas (definidas en las ecuaciones señaladas previamente), que se relacionan con cada una de las velocidades lineales y angulares de los puntos de análisis, lo que permitiría crear una matriz de inercia, sin embargo, debido a la complejidad y redundancia de este tipo de robots, **es recomendable** definir la matriz de inercia después de evaluar cada una de las derivadas en la ecuación (5.27). Mientras tanto, la ecuación de energía cinética puede reducirse a

$$K_{j/0}^0 = \frac{1}{2} \sum_{j=1}^3 \left[(m_l + m_v) \dot{r}_{j/0}^T \dot{r}_{j/0}^0 + m_v \dot{r}_{p_j/0}^T \dot{r}_{p_j/0}^0 + \omega_{j/0}^T \left(\mathbf{I}_{m/0}^0 + \mathbf{I}_{l/0}^0 \right) \omega_{j/0}^0 \right] + \frac{1}{2} m_e \dot{r}_{e/0}^T \dot{r}_{e/0}^0 \quad (5.16)$$

Esta simplificación es válida porque se consideró que todas las masas m_{l_i} y m_{v_i} son iguales y se expresan como m_l y m_v , respectivamente. Además, la velocidad angular de los motores $\omega_{m_j/0}^0$ serán iguales a sus homólogas en los cuerpos rígidos l_i , es decir, $\omega_{m_j/0}^0 = \omega_{l_j/0}^0$. El resultado depende de $\mathbf{I}_{m_j/0}^0, \mathbf{I}_{l_j/0}^0 \in \mathbb{R}^{3 \times 3}$, que son los tensores de inercia del tren de transmisión del motor y del cuerpo rígido acoplado a él, los cuales se explicarán en secciones posteriores. Por otro lado, la energía potencial depende de las ecuaciones (2.8), (2.38) y (5.13), esta se expresa y reduce de la siguiente manera:

$$\begin{aligned}
P_{j/0}^0 &= -g^T \left(m_{l_1} \bar{r}_{1/0}^0 + m_{l_2} \bar{r}_{2/0}^0 + m_{l_3} \bar{r}_{3/0}^0 + m_{v_1} \bar{r}_{p_1/0}^0 + m_{v_2} \bar{r}_{p_2/0}^0 + m_{v_3} \bar{r}_{p_3/0}^0 + m_e \bar{r}_{e/0}^0 \right) \\
&= -g^T \left[\sum_{j=1}^3 \left(m_l \bar{r}_{j/0}^0 + m_v \bar{r}_{p_j/0}^0 \right) + m_e \bar{r}_{e/0}^0 \right]. \quad (5.17)
\end{aligned}$$

El signo negativo en la ecuación anterior indica que los cuerpos rígidos en el sistema se encuentran **por debajo del marco de referencia inercial durante todo el ciclo de trabajo del robot**, sin embargo, la aceleración gravitacional también se debe definir en sentido contrario al marco inercial, esto es

$$g = \begin{bmatrix} 0 \\ 0 \\ -9.80665 \end{bmatrix} \quad (5.18)$$

Los resultados en las ecuaciones (5.16) y (5.17) serán utilizados en las secciones 5.4 y 5.4.1, pero antes de abordar esos temas, es importante definir los tensores de inercia de los elementos que conforman al robot.

5.3. Matrices de Rotación y Tensores de Inercia de los Cuerpos Rígidos

Un tensor de inercia $I_{\bullet} \in \mathbb{R}^{3 \times 3}$ describe, matemáticamente, la manera en la que la masa se distribuye en un cuerpo rígido respecto a su centro de masa \bullet [15]. En sistemas dinámicos que operan en espacios tridimensionales (e.g. robots paralelos tipo delta), el cálculo **simbólico** de un tensor de inercia puede ser muy complejo, pero existen herramientas de dibujo paramétrico que lo calculan **numéricamente**, sin embargo, estas calculan los tensores como si todas las piezas estuvieran alineadas con el marco de referencia inercial, i.e. como si cada centro de masa estuviera colocado en la base del robot durante todo el ciclo de trabajo (figura 5.5).

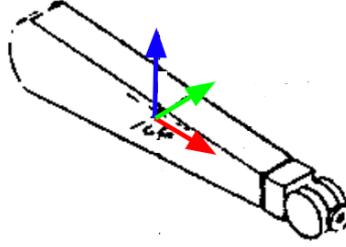


Figura 5.5: Ejemplo de un Cuerpo Rígido alineado con el Marco de Referencia Inercial.

Lo anterior, en robots **reales**, es incongruente, ya que cada centro de masa de cada cuerpo rígido se desplaza hacia diferentes direcciones respecto al marco inercial, lo que cambia su orientación por completo. Para resolver esta situación, se puede realizar una transformación entre marcos de referencia, la cual se describe como sigue

$$I_{\bullet/0}^0 = R_{\bullet/0}^0 I_{\bullet} R_{\bullet/0}^{-1} = R_{\bullet/0}^0 I_{\bullet} R_{\bullet/0}^T \quad (5.19)$$

donde $R_{\bullet/0}^0 \in \mathbb{R}^{3 \times 3}$ representa la matriz de rotación del cuerpo rígido estudiado, la cual se obtiene de las matrices de transformación homogénea de la cinemática directa hacia cada centro de masa y cuya forma inversa $R_{\bullet/0}^{-1}$ es igual a su forma transpuesta $R_{\bullet/0}^T$. Su significado físico se puede interpretar de la siguiente manera (figura 5.6):

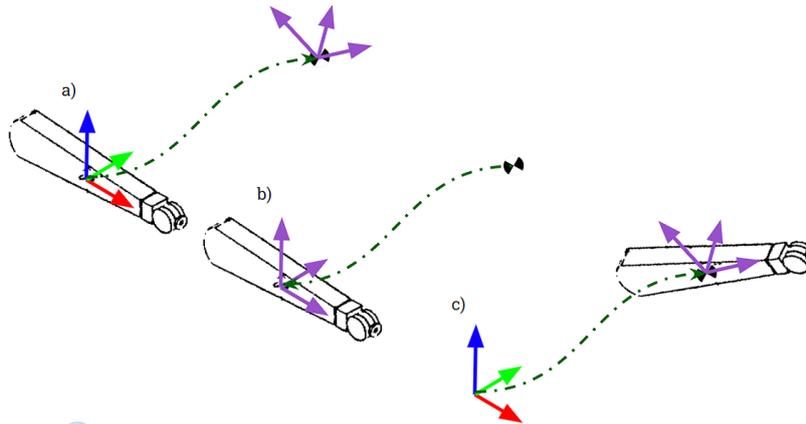


Figura 5.6: Ejemplo de Transformación entre Marcos de Referencia.

- a) A través de la cinemática directa, se conoce la **orientación** $R_{\bullet/0}^0$ del **centro de masa** respecto al marco **inercial**, donde los valores del tensor de inercia I_{\bullet} , calculados por software, se encuentran también
- b) El **marco del centro de masa** se alinea con el marco **inercial** al invertir la matriz de rotación $(R_{\bullet/0}^{-1})$. En la figura se ejemplifica como una traslación, pero en realidad **no se desplaza**, solo rota
- c) Estando alineado el **marco del centro de masa** con el marco **inercial**, se toma al tensor de inercia I_{\bullet} y se rota en el espacio con la matriz original $R_{\bullet/0}^0$. Al igual que en el caso anterior, en la figura se ejemplifica como una traslación, pero en realidad **no se desplaza**, solo rota

Los tensores de inercia $I_{\bullet/0}^0, I_{\bullet}$ de los cuerpos rígidos que constituyen un robot pueden cambiar de acuerdo a su masa y geometría, por lo que su cálculo dependerá de cada sistema, sin embargo, se generaliza su cálculo considerando al tensor de inercia, obtenido numéricamente, como

$$I_{\bullet} = \begin{bmatrix} I_{xx} & -I_{xy} & -I_{xz} \\ -I_{xy} & I_{yy} & -I_{yz} \\ -I_{xz} & -I_{yz} & I_{zz} \end{bmatrix}. \quad (5.20)$$

En el análisis de posición y velocidad para el cálculo de las energías cinética y potencial, se estudiaron los diferentes puntos de unión entre cuerpos rígidos y no sus centros de masa, por lo que los valores en el tensor de inercia cambian, sin embargo, estos se pueden transformar tal como se explicó anteriormente, solo que las matrices de rotación se definen en función del punto de análisis involucrado.

5.3.1. Cuerpos Rígidos l_i

Para el análisis de este tipo de robot, se considera que cada cuerpo rígido l_i tiene un tensor de inercia I_{l_i} , el cual se define de la siguiente manera:

$$I_{l_i} = \begin{bmatrix} I_{xx} & -I_{xy} & -I_{xz} \\ -I_{xy} & I_{yy} & -I_{yz} \\ -I_{xz} & -I_{yz} & I_{zz} \end{bmatrix}. \quad (5.21)$$

Su **postura** (posición y orientación) fue descrita en la ecuación (2.7), esto permite definir a la matriz de rotación como

$$R_{l_i/0}^0 = \begin{bmatrix} \cos(\phi_i) & -\sin(\phi_i)\cos(\theta_i) & -\sin(\phi_i)\sin(\theta_i) \\ \sin(\phi_i) & \cos(\phi_i)\cos(\theta_i) & \cos(\phi_i)\sin(\theta_i) \\ 0 & -\sin(\theta_i) & \cos(\theta_i) \end{bmatrix}. \quad (5.22)$$

Ya que calcular estas matrices de rotación no es algo complejo, se omite su algoritmo, pero el código en MATLAB puede encontrarse en el apéndice A.10. Con esto, el tensor de inercia I_{l_i} puede transformarse con las ecuaciones (5.19) y (5.22), sin embargo, debido a la extensión del resultado general (i.e. aquél que puede utilizarse para cualquiera de los tres cuerpos l_i), solo se mostrarán los tensores para cada cuerpo rígido considerando que $\phi_i = 0, \frac{2}{3}\pi, \frac{4}{3}\pi$, con $i = 1, 2, 3$:

$$I_{l_i/0}^0 = \begin{bmatrix} I_{xx_1} & I_{xz_1} \sin(\theta_1) - I_{xy_1} \cos(\theta_1) & -I_{xz_1} \cos(\theta_1) - I_{xy_1} \sin(\theta_1) \\ I_{xz_1} \sin(\theta_1) - I_{xy_1} \cos(\theta_1) & I_{yy_1} \cos^2(\theta_1) + 2I_{yz_1} \cos(\theta_1) \sin(\theta_1) + I_{zz_1} \sin^2(\theta_1) & \frac{1}{2} [I_{yy_1} - I_{zz_1}] \sin(2\theta_1) - I_{yz_1} \cos(2\theta_1) \\ -I_{xz_1} \cos(\theta_1) - I_{xy_1} \sin(\theta_1) & \frac{1}{2} [I_{yy_1} - I_{zz_1}] \sin(2\theta_1) - I_{yz_1} \cos(2\theta_1) & I_{yy_1} \sin^2(\theta_1) - 2I_{yz_1} \cos(\theta_1) \sin(\theta_1) + I_{zz_1} \cos^2(\theta_1) \end{bmatrix}, \quad (5.23)$$

$$I_{i/0}^0 = \frac{1}{2} \begin{bmatrix} \frac{1}{2}I_{xx2} + \frac{1}{2}I_{yy2} + \sqrt{3}I_{yz2} & \left(\frac{\sqrt{3}}{2}I_{xx2} + I_{yy2} - \frac{\sqrt{3}}{2}I_{yz2}\right) \cos(\theta_2) - (I_{xz2} - \sqrt{3}I_{yz2}) \sin(\theta_2) & (I_{xz2} - \sqrt{3}I_{yz2}) \cos(\theta_2) + \left(\frac{\sqrt{3}}{2}I_{xx2} + I_{yy2} - \frac{\sqrt{3}}{2}I_{yz2}\right) \sin(\theta_2) \\ \left(\frac{\sqrt{3}}{2}I_{xx2} + I_{yy2} - \frac{\sqrt{3}}{2}I_{yz2}\right) \cos(\theta_2) - (I_{xz2} - \sqrt{3}I_{yz2}) \sin(\theta_2) & \left(\frac{1}{2}I_{xx2} - \sqrt{3}I_{yy2} + \frac{1}{2}I_{yz2}\right) \cos(\theta_2)^2 - (\sqrt{3}I_{xz2} + I_{yz2}) \sin(2\theta_2) + 2I_{xz2} \sin(\theta_2)^2 & (\sqrt{3}I_{xz2} + I_{yz2}) \cos(2\theta_2) + \left(\frac{1}{2}I_{xx2} - \frac{\sqrt{3}}{2}I_{yy2} + \frac{1}{2}I_{yz2} - I_{xz2}\right) \sin(2\theta_2) \\ (I_{xz2} - \sqrt{3}I_{yz2}) \cos(\theta_2) + \left(\frac{\sqrt{3}}{2}I_{xx2} + I_{yy2} - \frac{\sqrt{3}}{2}I_{yz2}\right) \sin(\theta_2) & (\sqrt{3}I_{xz2} + I_{yz2}) \cos(2\theta_2) + \left(\frac{1}{2}I_{xx2} - \frac{\sqrt{3}}{2}I_{yy2} + \frac{1}{2}I_{yz2} - I_{xz2}\right) \sin(2\theta_2) & \frac{1}{2}(3I_{xx2} + I_{yy2} - 2\sqrt{3}I_{yz2}) \sin(\theta_2)^2 + (\sqrt{3}I_{xz2} + I_{yz2}) \sin(2\theta_2) + 2I_{xz2} \cos(\theta_2)^2 \end{bmatrix}, \quad (5.24)$$

$$I_{i/0}^0 = \frac{1}{2} \begin{bmatrix} \frac{1}{2}I_{xx2} + \frac{1}{2}I_{yy2} - \sqrt{3}I_{yz2} & \left(-\frac{\sqrt{3}}{2}I_{xx2} - I_{yy2} + \frac{\sqrt{3}}{2}I_{yz2}\right) \cos(\theta_2) - (I_{xz2} + \sqrt{3}I_{yz2}) \sin(\theta_2) & (I_{xz2} + \sqrt{3}I_{yz2}) \cos(\theta_2) - \left(\frac{\sqrt{3}}{2}I_{xx2} - I_{yy2} + \frac{\sqrt{3}}{2}I_{yz2}\right) \sin(\theta_2) \\ \left(-\frac{\sqrt{3}}{2}I_{xx2} - I_{yy2} + \frac{\sqrt{3}}{2}I_{yz2}\right) \cos(\theta_2) - (I_{xz2} + \sqrt{3}I_{yz2}) \sin(\theta_2) & \left(\frac{1}{2}I_{xx2} + \sqrt{3}I_{yy2} + \frac{1}{2}I_{yz2}\right) \cos(\theta_2)^2 - (\sqrt{3}I_{xz2} + I_{yz2}) \sin(2\theta_2) + 2I_{xz2} \sin(\theta_2)^2 & (\sqrt{3}I_{xz2} + I_{yz2}) \cos(2\theta_2) + \left(\frac{1}{2}I_{xx2} + \frac{\sqrt{3}}{2}I_{yy2} + \frac{1}{2}I_{yz2} - I_{xz2}\right) \sin(2\theta_2) \\ (I_{xz2} + \sqrt{3}I_{yz2}) \cos(\theta_2) - \left(\frac{\sqrt{3}}{2}I_{xx2} - I_{yy2} + \frac{\sqrt{3}}{2}I_{yz2}\right) \sin(\theta_2) & (-\sqrt{3}I_{xz2} + I_{yz2}) \cos(2\theta_2) + \left(\frac{1}{2}I_{xx2} + \frac{\sqrt{3}}{2}I_{yy2} + \frac{1}{2}I_{yz2} - I_{xz2}\right) \sin(2\theta_2) & \frac{1}{2}(3I_{xx2} + I_{yy2} + 2\sqrt{3}I_{yz2}) \sin(\theta_2)^2 - (\sqrt{3}I_{xz2} - I_{yz2}) \sin(2\theta_2) + 2I_{xz2} \cos(\theta_2)^2 \end{bmatrix}, \quad (5.25)$$

Nótese la extensión de los resultados anteriores, por lo que se sugiere calcular los tensores de inercia (sin importar el cuerpo rígido estudiado) con MATLAB utilizando el código en el apéndice A.11. Se omite el algoritmo ya que solo se necesita que sus parámetros de entrada sean el tensor de inercia I_j de cualquier cuerpo rígido (calculado **numéricamente** por alguna herramienta de dibujo paramétrico) y su respectiva matriz de rotación $R_{j/0}^0$. **Se debe aclarar que el tensor de inercia de estos cuerpos rígidos se debe calcular respecto a su base y no a su centro de masa.** Por ejemplo, en [13] se propone que este tipo de cuerpos rígidos sean **varillas** (figura 5.7), cuyo tensor de inercia respecto a su base se describe como

$$I_{l_i} = \begin{bmatrix} \frac{1}{3}m_{l_i}l_i^2 & 0 & 0 \\ 0 & \frac{1}{3}m_{l_i}l_i^2 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (5.26)$$

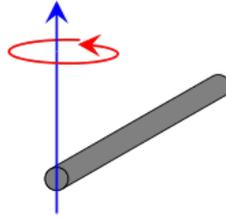


Figura 5.7: Ejemplo de la Simplificación en la Geometría del Eslabón l_i .

Sin embargo, los robots **reales** tienen cuerpos rígidos con geometrías más complejas, por lo que se sugiere calcular sus tensores con herramientas de dibujo paramétrico como SolidWorks.

5.4. Modelo Matemático

El modelo matemático de un sistema dinámico con múltiples variables se puede calcular con las ecuaciones de Euler - Lagrange, y aunque es común definir las individualmente para cada articulación en un robot, en [11] se propone calcular el modelo de forma generalizada con la siguiente ecuación

$$\frac{d}{dt} \left(\frac{\partial L_{j/0}^0}{\partial \dot{\theta}} \right)^T - \left(\frac{\partial L_{j/0}^0}{\partial \theta} \right)^T = \tau \quad (5.27)$$

donde $L_{j/0}^0 \in \mathbb{R}$ es el lagrangiano en función de los puntos de análisis de velocidad dentro del robot, mientras que $\tau \in \mathbb{R}^{3 \times 1}$ es el torque aplicado en cada una de las articulaciones $\bar{\theta} \in \mathbb{R}^{3 \times 1}$ de un robot. El lagrangiano $L_{j/0}^0 \in \mathbb{R}$, que es un valor **escalar**, se calcula como la diferencia entre la energía cinética y potencial del sistema

$$L_{j/0}^0 = K_{j/0}^0 - P_{j/0}^0 \quad (5.28)$$

siendo $K_{j/0}^0$ y $P_{j/0}^0$ las energías cinética y potencial definidas en las ecuaciones (5.16) y (5.17). Debido a que existe una interrelación compleja entre los movimientos en las articulaciones y el

efector final, se debe calcular un modelo auxiliar que permita conectar ambos espacios de coordenadas en un solo modelo dinámico; esto se logra calculando el mismo lagrangiano de la ecuación (5.28), solo que la ecuación de Euler - Lagrange cambia de la siguiente manera

$$\frac{d}{dt} \left(\frac{\partial L_{j/0}^0}{\partial \dot{\vec{r}}_{e/0}^0} \right)^T - \left(\frac{\partial L_{j/0}^0}{\partial \vec{r}_{e/0}^0} \right)^T = \mathbf{f} \quad (5.29)$$

en este caso, $\mathbf{f} \in \mathbb{R}^{3 \times 1}$ es una fuerza que afecta directamente al efector final, que puede ser la misma generada por el robot o una fuerza externa. La diferencia entre ambas ecuaciones es que las derivadas parciales se realizan en función de las posiciones y velocidades de las articulaciones $\vec{\theta}, \dot{\vec{\theta}} \in \mathbb{R}^{3 \times 1}$ y de la posición y velocidad del efector final $\vec{r}_{e/0}^0, \dot{\vec{r}}_{e/0}^0 \in \mathbb{R}^{3 \times 1}$.

5.4.1. Restricciones Cinemáticas

Una forma muy simple para definir la relación entre modelos dinámicos es **estableciendo una restricción $S_i \in \mathbb{R}$ en el sistema, la cual conecta directamente el movimiento de las articulaciones con los movimientos en el efector final**. Esto se había hecho anteriormente con las ecuaciones (2.11) y (2.12), que definen tres esferas que acotan el comportamiento **cinemático** del efector final; en este caso tienen la misma función, solo que para este análisis, las ecuaciones se estructuran en forma matricial de la siguiente manera

$$\begin{aligned} S_i &= \left(x - c_{x_{i/0}}^0 \right)^2 + \left(y - c_{y_{i/0}}^0 \right)^2 + \left(z - c_{z_{i/0}}^0 \right)^2 - v_i^2 = 0 \\ &= \left(\begin{bmatrix} x - c_{x_{i/0}}^0 & y - c_{y_{i/0}}^0 & z - c_{z_{i/0}}^0 \end{bmatrix} \right) \left(\begin{bmatrix} x - c_{x_{i/0}}^0 \\ y - c_{y_{i/0}}^0 \\ z - c_{z_{i/0}}^0 \end{bmatrix} \right) - v_i^2 = 0 \\ &= \left(\vec{r}_{e/0}^0 - \vec{c}_{v_{i/0}}^0 \right)^T \left(\vec{r}_{e/0}^0 - \vec{c}_{v_{i/0}}^0 \right) - v_i^2 = 0, \end{aligned} \quad (5.30)$$

con $i \in \mathbb{R}, i = 1, 2, 3$, que representa a la i -ésima articulación. El significado físico de esta restricción es que se delimita el espacio de trabajo del efector final, el cual solo se desplaza dentro del espacio en el que las esferas se intersecan (figura 2.5), lo que directamente limita el movimiento de las articulaciones, evitando posiciones singulares. Para incorporarlo en las ecuaciones (5.27) y (5.29), la restricción se estudia con multiplicadores de Lagrange, derivándola respecto a las coordenadas generalizadas $\vec{\theta}$ y $\vec{r}_{e/0}^0$, esto es

$$\sum_{k=1}^3 \lambda_k \left(\frac{\partial S_i}{\partial \theta_k} \right) = 2 \sum_{k=1}^3 \lambda_k \left(\vec{r}_{e/0}^0 - \vec{c}_{v_{i/0}}^0 \right)^T \left(\frac{\partial \vec{r}_{e/0}^0}{\partial \theta_k} - \frac{\partial \vec{c}_{v_{i/0}}^0}{\partial \theta_k} \right), \quad (5.31)$$

$$\sum_{k=1}^3 \lambda_k \left(\frac{\partial S_i}{\partial r_k} \right) = 2 \sum_{k=1}^3 \lambda_k \left(\vec{r}_{e/0}^0 - \vec{c}_{v_{i/0}}^0 \right)^T \left(\frac{\partial \vec{r}_{e/0}^0}{\partial r_k} - \frac{\partial \vec{c}_{v_{i/0}}^0}{\partial r_k} \right), \quad (5.32)$$

donde $\vec{c}_{v_{i/0}}^0 \in \mathbb{R}^{3 \times 1}$ es el vector de posición del i -ésimo centro instantáneo de rotación, relacionado con la i -ésima articulación y definido en la ecuación (2.11); asimismo, $\frac{\partial \vec{r}_{e/0}^0}{\partial \theta_k}, \frac{\partial \vec{c}_{v_{i/0}}^0}{\partial \theta_k}, \frac{\partial \vec{r}_{e/0}^0}{\partial r_k}, \frac{\partial \vec{c}_{v_{i/0}}^0}{\partial r_k} \in \mathbb{R}^{3 \times 1}$ son:

1. $\frac{\partial \vec{r}_{e/0}^0}{\partial \theta_k}$: k -ésima columna de la matriz jacobiana geométrica $J^I(\vec{n}, \vec{r})$, ecuaciones (3.5) y (3.22)

2. $\frac{\partial \bar{c}_{v_i/0}^0}{\partial \theta_k}$: derivada del i -ésimo centro instantáneo de rotación respecto a la k -ésima articulación
3. $\frac{\partial \bar{r}_{e/0}^0}{\partial r_k}$: derivada de la posición del efector final respecto a sus propias coordenadas (x, y, z) . El valor de k representa a cada coordenada, e.g. $k = 1 = x, k = 2 = y, k = 3 = z$
4. $\frac{\partial \bar{c}_{v_i/0}^0}{\partial r_k}$: derivada de la posición del centro instantáneo de rotación respecto a las coordenadas (x, y, z) del efector final. El valor de k representa a cada coordenada, e.g. $k = 1 = x, k = 2 = y, k = 3 = z$

Exceptuando a $\frac{\partial \bar{r}_{e/0}^0}{\partial \theta_k}$, cuyo valor se puede obtener de la ecuación (3.26), cada una de las matrices listadas se calcula de la siguiente manera:

$$\begin{aligned} \frac{\partial \bar{c}_{v_i/0}^0}{\partial \theta_k} &= \frac{\partial}{\partial \theta_k} \left(\begin{bmatrix} \sin(\phi_i) [a_i - p_i + l_i \sin(\theta_i)] \\ -\cos(\phi_i) [a_i - p_i + l_i \sin(\theta_i)] \\ -l_i \cos(\theta_i) \end{bmatrix} \right) \\ &= \begin{bmatrix} l_i \sin(\phi_i) \cos(\theta_i) \\ -l_i \cos(\phi_i) \cos(\theta_i) \\ l_i \sin(\theta_i) \end{bmatrix} \cdot \frac{\partial \theta_i}{\partial \theta_k} = \left[(R_{\phi_i, z} \bar{n}_{a_i/0}^0) \times \bar{r}_{i/a_i}^0 \right] \frac{\partial \theta_i}{\partial \theta_k}, \end{aligned} \quad (5.33)$$

en este caso, $\frac{\partial \theta_i}{\partial \theta_k}$ será igual a 1 cuando $i = k$, de otra forma será igual a cero. La siguiente derivada a calcular es $\frac{\partial \bar{r}_{e/0}^0}{\partial r_k}$:

$$\begin{aligned} \frac{\partial \bar{r}_{e/0}^0}{\partial r_1} &= \frac{\partial}{\partial x} \left(\begin{bmatrix} x \\ y \\ z \end{bmatrix} \right) = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \\ \frac{\partial \bar{r}_{e/0}^0}{\partial r_2} &= \frac{\partial}{\partial y} \left(\begin{bmatrix} x \\ y \\ z \end{bmatrix} \right) = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \\ \frac{\partial \bar{r}_{e/0}^0}{\partial r_3} &= \frac{\partial}{\partial z} \left(\begin{bmatrix} x \\ y \\ z \end{bmatrix} \right) = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}. \end{aligned} \quad (5.34)$$

Estos resultados se pueden expresar como $1_k \in \mathbb{R}^{3 \times 1}$, donde el subíndice indica la fila donde existe el valor 1, el resto será igual a cero. Por último, la derivada $\frac{\partial \bar{c}_{v_i/0}^0}{\partial r_k}$ es:

$$\begin{aligned} \frac{\partial \bar{c}_{v_i/0}^0}{\partial r_1} &= \frac{\partial}{\partial x} \left(\begin{bmatrix} \sin(\phi_i) [a_i - p_i + l_i \sin(\theta_i)] \\ -\cos(\phi_i) [a_i - p_i + l_i \sin(\theta_i)] \\ -l_i \cos(\theta_i) \end{bmatrix} \right) \\ \frac{\partial \bar{c}_{v_i/0}^0}{\partial r_2} &= \frac{\partial}{\partial y} \left(\begin{bmatrix} \sin(\phi_i) [a_i - p_i + l_i \sin(\theta_i)] \\ -\cos(\phi_i) [a_i - p_i + l_i \sin(\theta_i)] \\ -l_i \cos(\theta_i) \end{bmatrix} \right) \\ \frac{\partial \bar{c}_{v_i/0}^0}{\partial r_3} &= \frac{\partial}{\partial z} \left(\begin{bmatrix} \sin(\phi_i) [a_i - p_i + l_i \sin(\theta_i)] \\ -\cos(\phi_i) [a_i - p_i + l_i \sin(\theta_i)] \\ -l_i \cos(\theta_i) \end{bmatrix} \right) \\ &= \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \end{aligned} \quad (5.35)$$

por lo tanto, las ecuaciones (5.31) y (5.32) se reducen a

$$\sum_{k=1}^3 \lambda_k \left(\frac{\partial S_i}{\partial \theta_k} \right) = 2 \sum_{k=1}^3 \lambda_k \left(\vec{r}_{e/0}^0 - \vec{c}_{v_i/0}^0 \right)^T \left(J^I(:, k) - \left[\left(R_{\phi_i, z} \vec{n}_{a_i/0}^0 \right) \times \vec{r}_{i/a_i}^0 \right] \frac{\partial \theta_i}{\partial \theta_k} \right), \quad (5.36)$$

$$\sum_{k=1}^3 \lambda_k \left(\frac{\partial S_i}{\partial r_k} \right) = 2 \sum_{k=1}^3 \lambda_k \left(\vec{r}_{e/0}^0 - \vec{c}_{v_i/0}^0 \right)^T \mathbf{1}_k. \quad (5.37)$$

Estos resultados serán útiles una vez que los modelos dinámicos hayan sido definidos utilizando las ecuaciones de Euler - Lagrange.

5.4.2. Modelo en el Espacio de las Articulaciones

Comenzando por el modelo dinámico de las articulaciones, su lagrangiano se puede definir utilizando las ecuaciones (5.16), (5.17) y (5.28) tal como se muestra a continuación:

$$L_{j/0}^0 = \frac{1}{2} \sum_{j=1}^3 \left[(m_l + m_v) \dot{\vec{r}}_{j/0}^0 \dot{\vec{r}}_{j/0}^0 + m_v \dot{\vec{r}}_{p_j/0}^0 \dot{\vec{r}}_{p_j/0}^0 + \omega_{j/0}^T \left(\Gamma_{m_j/0}^0 + \Gamma_{l_j/0}^0 \right) \omega_{j/0}^0 \right] + \frac{1}{2} m_e \dot{\vec{r}}_{e/0}^0 \dot{\vec{r}}_{e/0}^0 + g^T \left[\sum_{j=1}^3 (m_l \vec{r}_{j/0}^0 + m_v \vec{r}_{p_j/0}^0) + m_e \vec{r}_{e/0}^0 \right], \quad (5.38)$$

a partir de aquí solo se deben realizar las derivadas de la ecuación (5.27) de la siguiente manera:

$$\left(\frac{\partial L_{j/0}^0}{\partial \theta} \right)^T = \left[\sum_{j=1}^3 \left(m_l \cdot \frac{\partial \vec{r}_{j/0}^0}{\partial \theta} + m_v \cdot \frac{\partial \vec{r}_{p_j/0}^0}{\partial \theta} \right) + m_e \cdot \frac{\partial \vec{r}_{e/0}^0}{\partial \theta} \right]^T g, \quad (5.39)$$

esto puede simplificarse ya que los términos $\frac{\partial \vec{r}_{e/0}^0}{\partial \theta}$, $\frac{\partial \vec{r}_{j/0}^0}{\partial \theta}$, $\frac{\partial \vec{r}_{p_j/0}^0}{\partial \theta} \in \mathbb{R}^{3 \times 3}$ son las matrices jacobianas de velocidad lineal definidas en las ecuaciones (3.26), (5.8) a (5.10) y (5.12):

$$\begin{aligned} \left(\frac{\partial L_{j/0}^0}{\partial \theta} \right)^T &= \left[\sum_{j=1}^3 \left(m_l J_{v_j}^T + m_v J^T \right) + m_e J^T \right] g \\ &= \left[\sum_{j=1}^3 \left(m_l J_{v_j}^T \right) + (m_e + 3m_v) J^T \right] g \end{aligned} \quad (5.40)$$

El siguiente término por calcular es la derivada del lagrangiano en función de la velocidad de las articulaciones, esto es:

$$\left(\frac{\partial L_{j/0}^0}{\partial \dot{\theta}} \right)^T = \sum_{j=1}^3 \left[(m_l + m_v) \left(\frac{\partial \dot{\vec{r}}_{j/0}^0}{\partial \dot{\theta}} \right)^T \dot{\vec{r}}_{j/0}^0 + m_v \left(\frac{\partial \dot{\vec{r}}_{p_j/0}^0}{\partial \dot{\theta}} \right)^T \dot{\vec{r}}_{p_j/0}^0 + \left(\frac{\partial \omega_{j/0}^0}{\partial \dot{\theta}} \right)^T \left(\Gamma_{m_j/0}^0 + \Gamma_{l_j/0}^0 \right) \omega_{j/0}^0 \right] + m_e \left(\frac{\partial \dot{\vec{r}}_{e/0}^0}{\partial \dot{\theta}} \right)^T \dot{\vec{r}}_{e/0}^0, \quad (5.41)$$

donde las derivadas $\frac{\partial \dot{\vec{r}}_{j/0}^0}{\partial \dot{\theta}}$, $\frac{\partial \dot{\vec{r}}_{p_j/0}^0}{\partial \dot{\theta}}$, $\frac{\partial \omega_{j/0}^0}{\partial \dot{\theta}}$, $\frac{\partial \dot{\vec{r}}_{e/0}^0}{\partial \dot{\theta}} \in \mathbb{R}^{3 \times 3}$ se calculan de la siguiente manera:

$$\frac{\partial \dot{\vec{r}}_{j/0}^0}{\partial \dot{\theta}} = \left[\begin{array}{c} l_i \sin(\phi_i) \cos(\theta_i) \\ -l_i \cos(\phi_i) \cos(\theta_i) \\ l_i \sin(\theta_i) \end{array} \right] \frac{\partial \dot{\theta}_i}{\partial \dot{\theta}_1} \left[\begin{array}{c} l_i \sin(\phi_i) \cos(\theta_i) \\ -l_i \cos(\phi_i) \cos(\theta_i) \\ l_i \sin(\theta_i) \end{array} \right] \frac{\partial \dot{\theta}_i}{\partial \dot{\theta}_2} \left[\begin{array}{c} l_i \sin(\phi_i) \cos(\theta_i) \\ -l_i \cos(\phi_i) \cos(\theta_i) \\ l_i \sin(\theta_i) \end{array} \right] \frac{\partial \dot{\theta}_i}{\partial \dot{\theta}_3}, \quad (5.42)$$

en este caso, las derivadas $\frac{\partial \dot{\theta}_i}{\partial \dot{\theta}_1}$, $\frac{\partial \dot{\theta}_i}{\partial \dot{\theta}_2}$, $\frac{\partial \dot{\theta}_i}{\partial \dot{\theta}_3} \in \mathbb{R}$ serán iguales a 1 cuando $\frac{\partial \dot{\theta}_1}{\partial \dot{\theta}_1}$, $\frac{\partial \dot{\theta}_2}{\partial \dot{\theta}_2}$, $\frac{\partial \dot{\theta}_3}{\partial \dot{\theta}_3}$, de otra forma, las derivadas serán iguales a cero. Por ejemplo, cuando $i = j = 2$, $\phi_2 = \frac{2}{3}\pi$, la derivada será

$$\frac{\partial \dot{\vec{r}}_{2/0}^0}{\partial \dot{\theta}} = \begin{bmatrix} 0 & \frac{\sqrt{3}}{2} l_2 \cos(\theta_2) & 0 \\ 0 & \frac{1}{2} l_2 \cos(\theta_2) & 0 \\ 0 & l_2 \sin(\theta_2) & 0 \end{bmatrix}, \quad (5.43)$$

que es igual a la matriz jacobiana de velocidad lineal $J_{v_2}(\vec{n}, \vec{r})$ de la ecuación (5.9), por lo tanto, la derivada $\frac{\partial \dot{\theta}_i}{\partial \dot{\theta}_1}$ es igual al resultado de la ecuación (5.7). Este mismo principio de análisis lleva a que el resto de derivadas se definen como

$$\frac{\partial \dot{r}_{p_j/0}^0}{\partial \dot{\theta}} = \frac{\partial \dot{r}_{e/0}^0}{\partial \dot{\theta}} = J^I(\vec{n}, \vec{r}), \text{ ecuación (3.26)} \quad (5.44)$$

$$\frac{\partial \omega_{j/0}^0}{\partial \dot{\theta}} = J_{\omega_j}(\vec{n}, \vec{r}), \text{ ecuaciones (5.4) a (5.6).} \quad (5.45)$$

Con esto se puede reducir la ecuación (5.41) de la siguiente manera:

$$\begin{aligned} \left(\frac{\partial L_{j/0}^0}{\partial \dot{\theta}} \right)^T &= \sum_{j=1}^3 \left[(m_l + m_v) J_{v_j}^T J_{v_j} \dot{\theta} + m_v J^T J \dot{\theta} + J_{\omega_j}^T \left(\mathbf{I}_{m_j/0}^0 + \mathbf{I}_{l_j/0}^0 \right) J_{\omega_j} \dot{\theta} \right] + m_e J^T J \dot{\theta}, \\ \left(\frac{\partial L_{j/0}^0}{\partial \dot{\theta}} \right)^T &= \underbrace{\left(\sum_{j=1}^3 \left[(m_l + m_v) J_{v_j}^T J_{v_j} + J_{\omega_j}^T \left(\mathbf{I}_{m_j/0}^0 + \mathbf{I}_{l_j/0}^0 \right) J_{\omega_j} \right] + (m_e + 3m_v) J^T J \right)}_{\text{matriz de inercia } D(\bar{\theta}) \in \mathbb{R}^{3 \times 3}} \dot{\theta}, \end{aligned} \quad (5.46)$$

Por último, se calcula la derivada en función del tiempo del resultado anterior, lo que lleva al siguiente resultado

$$\frac{d}{dt} \left(\frac{\partial L_{j/0}^0}{\partial \dot{\theta}} \right)^T = \dot{D} \dot{\theta} + D \ddot{\theta}, \quad (5.47)$$

donde $\dot{D}(\bar{\theta}) \in \mathbb{R}^{3 \times 3}$ es la derivada de la matriz de inercia, que se puede calcular fácilmente derivando algunos términos de la ecuación (5.46)

$$\dot{D}(\bar{\theta}) = \left[\sum_{j=1}^3 \left[(m_l + m_v) \left(J_{v_j}^T J_{v_j} + J_{v_j}^T J_{v_j} \right) + J_{\omega_j}^T \left(\mathbf{I}_{m_j/0}^0 + \mathbf{I}_{l_j/0}^0 \right) J_{\omega_j} + J_{\omega_j}^T \left(\mathbf{I}_{m_j/0}^0 + \mathbf{I}_{l_j/0}^0 \right) \dot{J}_{\omega_j} \right] + (m_e + 3m_v) \left(J^T J + J^T \dot{J} \right) \right]. \quad (5.48)$$

En el resultador anterior aparece la derivada de la matriz jacobiana geométrica como \dot{J} , la cual se calcula con la ecuación (4.13); por otro lado, si se derivan las matrices jacobianas de las ecuaciones (5.4) a (5.6), se obtiene que cada $\dot{J}_{\omega_j} = 0_{3 \times 3}$, sin embargo, las matrices $\dot{J}_{v_j} \in \mathbb{R}^{3 \times 3}$ se calculan derivando las matrices jacobianas en las ecuaciones (5.8) a (5.10), que en general se debe derivar el término $\left(R_{\frac{4}{3}\pi, z} \vec{n}_{a_3/0}^0 \right) \times \vec{r}_{3/a_3}^0$ de la siguiente manera

$$\begin{aligned} \frac{d}{dt} \left[\left(R_{\phi_i, z} \vec{n}_{a_i/0}^0 \right) \times \vec{r}_{i/a_i}^0 \right] &= \underbrace{\frac{d}{dt} \left[\left(R_{\phi_i, z} \vec{n}_{a_i/0}^0 \right) \right]}_{=0_{3 \times 1}} \times \vec{r}_{i/a_i}^0 + \left(R_{\phi_i, z} \vec{n}_{a_i/0}^0 \right) \times \underbrace{\frac{d}{dt} \left(\vec{r}_{i/a_i}^0 \right)}_{\text{ecuación (4.6)}} \\ &= \left(R_{\phi_i, z} \vec{n}_{a_i/0}^0 \right) \times \left[\left(R_{\phi_i, z} \vec{n}_{a_i/0}^0 \right) \times \vec{r}_{i/a_i}^0 \right] \dot{\theta}_i, \end{aligned} \quad (5.49)$$

lo que permite definir a cada \dot{J}_{v_j} de la siguiente manera:

$$J_{v_1}^I = \left[\left(R_{0, z} \vec{n}_{a_1/0}^0 \right) \times \left[\left(R_{0, z} \vec{n}_{a_1/0}^0 \right) \times \vec{r}_{1/a_1}^0 \right] \dot{\theta}_1 \quad 0_{3 \times 1} \quad 0_{3 \times 1} \right] \quad (5.50)$$

$$J_{v_2}^I = \left[0_{3 \times 1} \quad \left(R_{\frac{2}{3}\pi, z} \vec{n}_{a_2/0}^0 \right) \times \left[\left(R_{\frac{2}{3}\pi, z} \vec{n}_{a_2/0}^0 \right) \times \vec{r}_{2/a_2}^0 \right] \dot{\theta}_2 \quad 0_{3 \times 1} \right] \quad (5.51)$$

$$\mathbf{j}_{v_3}^I = \begin{bmatrix} 0_{3 \times 1} & 0_{3 \times 1} & \left(R_{\frac{4}{3}\pi, z} \vec{n}_{a_3/0}^0 \right) \times \left[\left(R_{\frac{4}{3}\pi, z} \vec{n}_{a_3/0}^0 \right) \times \vec{r}_{3/a_3}^0 \right] \dot{\theta}_3 \end{bmatrix} \quad (5.52)$$

Sin embargo, cualquiera de estos resultados, al ser multiplicados por su respectiva matriz jacobiana de velocidad lineal, dará como resultado una matriz con ceros en todas sus casillas, para demostrarlo se realizará el producto $\mathbf{j}_{v_3}^T J_{v_3}^I$ (aunque se puede demostrar con cualquiera de las tres matrices):

$$\begin{aligned} \mathbf{j}_{v_3}^T J_{v_3}^I &= J_{v_3}^T \mathbf{j}_{v_3}^I = \begin{bmatrix} 0_{1 \times 3} \\ 0_{1 \times 3} \\ \left(R_{\frac{4}{3}\pi, z} \vec{n}_{a_3/0}^0 \right)^T \times \left[\left(R_{\frac{4}{3}\pi, z} \vec{n}_{a_3/0}^0 \right) \times \vec{r}_{3/a_3}^0 \right]^T \dot{\theta}_3 \end{bmatrix} \begin{bmatrix} 0_{3 \times 1} & 0_{3 \times 1} & \left(R_{\frac{4}{3}\pi, z} \vec{n}_{a_3/0}^0 \right) \times \vec{r}_{3/a_3}^0 \end{bmatrix} \\ &= \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \left(R_{\frac{4}{3}\pi, z} \vec{n}_{a_3/0}^0 \right)^T \times \left[\left(R_{\frac{4}{3}\pi, z} \vec{n}_{a_3/0}^0 \right) \times \vec{r}_{3/a_3}^0 \right]^T \left[\left(R_{\frac{4}{3}\pi, z} \vec{n}_{a_3/0}^0 \right) \times \vec{r}_{3/a_3}^0 \right] \end{bmatrix} \dot{\theta}_3 \\ &= \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \underbrace{\left[\left(R_{\frac{4}{3}\pi, z} \vec{n}_{a_3/0}^0 \right) \times \vec{r}_{3/a_3}^0 \right]^T}_{a^T} \left[\underbrace{\left(R_{\frac{4}{3}\pi, z} \vec{n}_{a_3/0}^0 \right)}_b \times \underbrace{\left[\left(R_{\frac{4}{3}\pi, z} \vec{n}_{a_3/0}^0 \right) \times \vec{r}_{3/a_3}^0 \right]}_a \right] \end{bmatrix} \dot{\theta}_3 \\ &= 0_{3 \times 3}, \end{aligned} \quad (5.53)$$

lo que es válido ya que existe un *triple producto escalar* [7] con dos términos similares, lo que convierte al término en un valor igual a cero. Con esto, la derivada de la matriz de inercia se reduce a lo siguiente:

$$\dot{D}(\bar{\theta}) = C(\bar{\theta}, \dot{\bar{\theta}}) = (m_e + 3m_v) \left(\mathbf{j}^T J + J^T \mathbf{j} \right), \quad (5.54)$$

donde $C(\bar{\theta}, \dot{\bar{\theta}}) \in \mathbb{R}^{3 \times 3}$ representa a la matriz de efectos centrífugos y de Coriolis de este tipo de robots. Al agrupar todas las derivadas calculadas previamente, se puede definir, **parcialmente**, el modelo dinámico del sistema

$$D(\bar{\theta}) \ddot{\bar{\theta}} + C(\bar{\theta}, \dot{\bar{\theta}}) \dot{\bar{\theta}} + \underbrace{\left[-\sum_{j=1}^3 (m_l J_{v_j}^T) + (m_e + 3m_v) J^T \right]}_{G(\bar{\theta})} g = \tau, \quad (5.55)$$

con $G(\bar{\theta}) \in \mathbb{R}^{3 \times 1}$, que es la matriz de efectos gravitacionales. Para calcular cada término en este modelo, se presentan los algoritmos 10 a 12, cuyos códigos en MATLAB se encuentran en los apéndices A.12 a A.14. Para comprender mejor el código, se considera que cada una de las masas m_l , m_v y m_e se envían, como parámetro de entrada de las funciones, en un **arreglo**, es decir

$$m = [m_l \quad m_v \quad m_e], \quad (5.56)$$

esto mismo se hace con los tensores de inercia del motor I_m y del cuerpo I_l , solo que estos se almacenan en una **variable tipo celda** de la siguiente manera:

$$I = \{I_m, I_l\}. \quad (5.57)$$

Algoritmo 10: Matriz de Inercia $D(\bar{\theta})$

Entrada: $(\bar{r}_{e/0}^0, \bar{\theta}, a, l, p, m, I)$; /* Parámetros geométricos */
 $J^I(\bar{n}, \bar{r}) \leftarrow$ ecuación (3.26); /* Matriz jacobiana geométrica */
 $[J_v, J_\omega] \leftarrow$ algoritmo 9; /* Matrices jacobianas de velocidad lineal y angular */
 $R_l \leftarrow$ apéndice A.11; /* Matrices de rotación de los cuerpos l_i */
 $D(\bar{\theta}) \leftarrow (m_e + 3m_v) J^T J$; /* Inicializa la matriz de inercia */
for $j \leftarrow 1$ **to** 3 **do**
 $I_{m_j/0}^0 \leftarrow R_{l_i}^T \{j\} I \{1\} R_{l_i} \{j\}$; /* Transforma al tensor del j -ésimo motor */
 $I_{l_j/0}^0 \leftarrow R_{l_i}^T \{j\} I \{2\} R_{l_i} \{j\}$; /* Transforma al tensor de l_j */
 $D(\bar{\theta}) \leftarrow D(\bar{\theta}) + (m_e + m_v) J_v^T \{j\} J_v \{j\} + J_\omega^T \{j\} (I_{m_j/0}^0 + I_{l_j/0}^0) J_\omega \{j\}$
end
Salida: $D(\bar{\theta})$

El caso de la matriz de efectos centrífugos y de Coriolis es algo fácil de resolver, ya que solo se deben utilizar la matriz jacobiana geométrica y su derivada en función del tiempo como parámetros de entrada, además de las masas de los cuerpos rígidos.

Algoritmo 11: Matriz de Efectos Centrífugos y de Coriolis $C(\bar{\theta}, \dot{\bar{\theta}})$

Entrada: (J^I, \dot{J}^I, m) ; /* Matrices y masas */
 $C(\bar{\theta}, \dot{\bar{\theta}}) \leftarrow (m_e + 3m_v) (J^T J + J^T \dot{J})$; /* Matriz de la ecuación (5.54) */
Salida: $C(\bar{\theta}, \dot{\bar{\theta}})$

Para calcular la matriz de efectos gravitacionales se debe recordar que **la aceleración gravitacional se define como en la ecuación (5.18) siempre que el marco de referencia inercia sea definido tal como se ha mostrado a lo largo de este documento.**

Algoritmo 12: Matriz de Efectos Gravitacionales $G(\bar{\theta})$

Entrada: $(\bar{r}_{e/0}^0, \bar{\theta}, a, l, p, m, g)$; /* Parámetros geométricos */
 $J^I(\bar{n}, \bar{r}) \leftarrow$ ecuación (3.26); /* Matriz jacobiana geométrica */
 $[J_v, \sim] \leftarrow$ algoritmo 9; /* Matrices jacobianas de velocidad lineal solamente */
 $G(\bar{\theta}) \leftarrow (m_e + 3m_v) J^T$; /* Inicializa la matriz de efectos gravitacionales */
for $j \leftarrow 1$ **to** 3 **do**
 $G(\bar{\theta}) \leftarrow G(\bar{\theta}) + m_l J_v^T \{j\}$
end
 $G(\bar{\theta}) \leftarrow -G(\bar{\theta}) g$; /* Multiplica por g y el valor calculado se hace negativo porque los cuerpos rígidos se encuentran por debajo del marco inercial en todo momento */
Salida: $G(\bar{\theta})$

El modelo dinámico en la ecuación (5.55) debe restringirse con los resultados de la ecuación (5.31) en el torque generalizado, ya que este es quien genera los movimientos de las articulaciones; esto se describe de mejor manera con la siguiente ecuación

$$D(\bar{\theta}) \ddot{\bar{\theta}} + C(\bar{\theta}, \dot{\bar{\theta}}) \dot{\bar{\theta}} + B\dot{\bar{\theta}} + G(\bar{\theta}) = \tau + 2 \begin{bmatrix} \sum_{k=1}^3 \lambda_k (\bar{r}_{e/0}^0 - \bar{c}_{v_1/0}^0)^T \left(J^I(:, k) - \left[(R_{0,z} \bar{n}_{a_1/0}^0) \times \bar{r}_{1/a_1}^0 \right] \frac{\partial \theta_1}{\partial \theta_k} \right) \\ \sum_{k=1}^3 \lambda_k (\bar{r}_{e/0}^0 - \bar{c}_{v_2/0}^0)^T \left(J^I(:, k) - \left[(R_{\frac{2}{3}\pi, z} \bar{n}_{a_2/0}^0) \times \bar{r}_{2/a_2}^0 \right] \frac{\partial \theta_2}{\partial \theta_k} \right) \\ \sum_{k=1}^3 \lambda_k (\bar{r}_{e/0}^0 - \bar{c}_{v_3/0}^0)^T \left(J^I(:, k) - \left[(R_{\frac{4}{3}\pi, z} \bar{n}_{a_3/0}^0) \times \bar{r}_{3/a_3}^0 \right] \frac{\partial \theta_3}{\partial \theta_k} \right) \end{bmatrix},$$

$$D(\bar{\theta}) \ddot{\bar{\theta}} + C(\bar{\theta}, \dot{\bar{\theta}}) \dot{\bar{\theta}} + B\dot{\bar{\theta}} + G(\bar{\theta}) = \tau + \tau_r, \quad (5.58)$$

donde $B \in \mathbb{R}^{3 \times 3}$ es la matriz con los coeficientes de rozamiento viscoso de cada una de las articulaciones del sistema y se incluye en el modelo para hacerlo lo más realista posible. Por otro lado, $\tau_r \in \mathbb{R}^{3 \times 1}$ representa al torque de restricción que evita las singularidades en el sistema. Este se puede calcular con el siguiente algoritmo, cuyo código en MATLAB se encuentra en el apéndice A.15.

Algoritmo 13: Torque de Restricción τ_r

```

Entrada:  $(\bar{r}_{e/0}^0, \bar{\theta}, a, l, p, \bar{\lambda})$ ; /* Parámetros geométricos */
 $\phi \leftarrow \frac{2}{3}\pi$ ; /* Separación entre articulaciones */
 $J^I(\bar{n}, \bar{r}) \leftarrow$  ecuación (3.26); /* Matriz jacobiana geométrica */
 $\tau_r \leftarrow$  zeros(3, 1); /* Inicializa el torque de restricción */

/* Itera sobre cada articulación */
for  $i \leftarrow 1$  to 3 do
     $\bar{c}_{v_i/0}^0 \leftarrow$  ecuación (2.11); /* Posición del  $i$ -ésimo centro de rotación */
     $(R_{\phi_i} \bar{n}_{a_i/0}^0) \times \bar{r}_{i/a_i}^0 \leftarrow$  ecuaciones (3.13) y (3.15)

    /* Itera sobre cada  $\lambda_k$  */
    for  $k \leftarrow 1$  to 3 do

        /* Si los subíndices coinciden */
        if  $i = k$  then
             $\tau_r(i) \leftarrow \tau_r(i) + 2\lambda_k (\bar{r}_{e/0}^0 - \bar{c}_{v_i/0}^0)^T \left[ J^I(:, k) - (R_{\phi_i} \bar{n}_{a_i/0}^0) \times \bar{r}_{i/a_i}^0 \right]$ ; /*  $\frac{\partial \theta_i}{\partial \theta_k} = 1$  */
        else
             $\tau_r(i) \leftarrow \tau_r(i) + 2\lambda_k (\bar{r}_{e/0}^0 - \bar{c}_{v_i/0}^0)^T [J^I(:, k)]$ ; /*  $\frac{\partial \theta_i}{\partial \theta_k} = 0$  */
        end
    end
end
Salida:  $\tau_r$ 

```

Este algoritmo requiere de los coeficientes $\bar{\lambda} \in \mathbb{R}^{3 \times 1}$, los cuales pueden calcularse a partir del modelo **auxiliar** del robot en el espacio cartesiano, es decir, utilizando las ecuaciones (5.29) y (5.37).

5.4.3. Modelo Auxiliar en el Espacio Cartesiano

La ecuación de energía cinética y el lagrangiano, para este modelo dinámico, se definen tal como se hizo en secciones previas (su tamaño tuvo que reducirse para poder mostrarlas detalladamente), solo que ahora no se expanden para obtener las velocidades de las articulaciones, es decir

$$L_{j/0}^0 = \frac{1}{2} \sum_{j=1}^3 \left[(m_l + m_v) \dot{r}_{j/0}^T \dot{r}_{j/0}^0 + m_v \dot{r}_{p_j/0}^T \dot{r}_{p_j/0}^0 + \omega_{j/0}^T \left(\mathbb{I}_{m/0}^0 + \mathbb{I}_{l/0}^0 \right) \omega_{j/0}^0 \right] + \frac{1}{2} m_e \dot{r}_{e/0}^T \dot{r}_{e/0}^0 + g^T \left[\sum_{j=1}^3 (m_l \bar{r}_{j/0}^0 + m_v \bar{r}_{p_j/0}^0) + m_e \bar{r}_{e/0}^0 \right] \quad (5.59)$$

A diferencia del análisis en el espacio de las articulaciones, aquí lo importante es trabajar con coordenadas cartesianas sin involucrar a las variables $\bar{\theta}, \dot{\bar{\theta}}$, entonces se puede comenzar a operar las derivadas del lagrangiano tal como se indica en la ecuación (5.29):

$$\left(\frac{\partial L_{j/0}^0}{\partial \bar{r}_{e/0}^0} \right)^T = \left[\sum_{j=1}^3 \left(m_v \cdot \frac{\partial \bar{r}_{p_j/0}^0}{\partial \bar{r}_{e/0}^0} \right) + m_e \cdot \frac{\partial \bar{r}_{e/0}^0}{\partial \bar{r}_{e/0}^0} \right]^T g \quad (5.60)$$

Para obtener el resultado anterior, se consideró que las derivadas de la mayor parte de términos son iguales a cero, ya que son velocidades (lineales y angulares) **que no dependen explícitamente de la posición del efector final** $\bar{r}_{e/0}^0$, sin embargo, la ecuación (5.13) contiene al vector $\bar{r}_{p_j/0}^0$, que si se expandiera en el lagrangiano que se está operando, podrían derivarse en función de las coordenadas (x, y, z) del vector $\bar{r}_{e/0}^0$. Entonces, $\frac{\partial \bar{r}_{p_j/0}^0}{\partial \bar{r}_{e/0}^0}, \frac{\partial \bar{r}_{e/0}^0}{\partial \bar{r}_{e/0}^0} \in \mathbb{R}^{3 \times 3}$ son:

$$\frac{\partial \bar{r}_{e/0}^0}{\partial \bar{r}_{e/0}^0} = \begin{bmatrix} \frac{\partial x}{\partial x} & \frac{\partial x}{\partial y} & \frac{\partial x}{\partial z} \\ \frac{\partial y}{\partial x} & \frac{\partial y}{\partial y} & \frac{\partial y}{\partial z} \\ \frac{\partial z}{\partial x} & \frac{\partial z}{\partial y} & \frac{\partial z}{\partial z} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \mathbb{I}_{3 \times 3} \quad (5.61)$$

$$\begin{aligned} \frac{\partial \bar{r}_{p_j/0}^0}{\partial \bar{r}_{e/0}^0} &= \begin{bmatrix} \frac{\partial}{\partial x} \begin{bmatrix} x + p_i \sin(\phi_i) \\ y - p_i \cos(\phi_i) \\ z \end{bmatrix} & \frac{\partial}{\partial y} \begin{bmatrix} x + p_i \sin(\phi_i) \\ y - p_i \cos(\phi_i) \\ z \end{bmatrix} & \frac{\partial}{\partial z} \begin{bmatrix} x + p_i \sin(\phi_i) \\ y - p_i \cos(\phi_i) \\ z \end{bmatrix} \end{bmatrix} \\ &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \mathbb{I}_{3 \times 3}, \end{aligned} \quad (5.62)$$

por lo tanto, la ecuación (5.60) se simplifica de la siguiente manera:

$$\begin{aligned} \left(\frac{\partial L_{j/0}^0}{\partial \bar{r}_{e/0}^0} \right)^T &= \left[\sum_{j=1}^3 (m_v) + m_e \right] g \\ &= (m_e + 3m_v) g \end{aligned} \quad (5.63)$$

El siguiente término por calcular es la derivada del lagrangiano en función de la velocidad del efector final:

$$\left(\frac{\partial L_{j/0}^0}{\partial \dot{\bar{r}}_{e/0}^0} \right)^T = \sum_{j=1}^3 \left[m_v \left(\frac{\partial \dot{\bar{r}}_{p_j/0}^0}{\partial \dot{\bar{r}}_{e/0}^0} \right)^T \dot{\bar{r}}_{p_j/0}^0 \right] + m_e \left(\frac{\partial \dot{\bar{r}}_{e/0}^0}{\partial \dot{\bar{r}}_{e/0}^0} \right)^T \dot{\bar{r}}_{e/0}^0 \quad (5.64)$$

Al igual que en la otra derivada de este modelo dinámico, se consideró que las derivadas de la mayor parte de términos son iguales a cero, ya que son velocidades que no dependen **explícitamente** de la velocidad del efector final. Asimismo, $\frac{\partial \dot{\bar{r}}_{p_j/0}^0}{\partial \dot{\bar{r}}_{e/0}^0}, \frac{\partial \dot{\bar{r}}_{e/0}^0}{\partial \dot{\bar{r}}_{e/0}^0} \in \mathbb{R}^{3 \times 3}$ son:

$$\frac{\partial \dot{\bar{r}}_{e/0}^0}{\partial \dot{\bar{r}}_{e/0}^0} = \frac{\partial \dot{\bar{r}}_{p_j/0}^0}{\partial \dot{\bar{r}}_{e/0}^0} = \begin{bmatrix} \frac{\partial \dot{x}}{\partial \dot{x}} & \frac{\partial \dot{x}}{\partial \dot{y}} & \frac{\partial \dot{x}}{\partial \dot{z}} \\ \frac{\partial \dot{y}}{\partial \dot{x}} & \frac{\partial \dot{y}}{\partial \dot{y}} & \frac{\partial \dot{y}}{\partial \dot{z}} \\ \frac{\partial \dot{z}}{\partial \dot{x}} & \frac{\partial \dot{z}}{\partial \dot{y}} & \frac{\partial \dot{z}}{\partial \dot{z}} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \mathbb{I}_{3 \times 3}, \quad (5.65)$$

esto es válido debido a la forma en la que se calcula la velocidad $\dot{\bar{r}}_{p_j/0}^0$, definida en la ecuación (5.12), lo que reduce la expresión anterior a

$$\left(\frac{\partial L_{j/0}^0}{\partial \dot{\bar{r}}_{e/0}^0} \right)^T = (m_e + 3m_v) \dot{\bar{r}}_{e/0}^0 \quad (5.66)$$

Por último, se calcula la derivada en función del tiempo del resultado anterior

$$\frac{d}{dt} \left(\frac{\partial L_{j/0}^0}{\partial \dot{\bar{r}}_{e/0}^0} \right)^T = (m_e + 3m_v) \ddot{\bar{r}}_{e/0}^0. \quad (5.67)$$

Al agrupar todas las derivadas calculadas previamente, se puede definir, **parcialmente**, el modelo auxiliar

$$(m_e + 3m_v) \ddot{\bar{r}}_{e/0}^0 - (m_e + 3m_v) g = \mathfrak{f} \quad (5.68)$$

donde $\ddot{\bar{r}}_{e/0}^0 \in \mathbb{R}^{3 \times 1}$ es la aceleración del efector final, la cual se calcula con las ecuaciones (3.26), (4.2) y (4.13) y el algoritmo 8. Con estos resultados no se busca crear otro modelo dinámico que funcione de forma independiente, sino utilizarlo para encontrar los valores de λ_i que serán utilizados en la ecuación (5.58). Por lo tanto, se considera que el modelo dinámico de la ecuación (5.68) se relaciona con la restricción de la ecuación (5.37) de la siguiente manera

$$(m_e + 3m_v) \ddot{\bar{r}}_{e/0}^0 - (m_e + 3m_v) g = \mathfrak{f} + 2 \begin{bmatrix} \sum_{k=1}^3 \lambda_k \left(\bar{r}_{e/0}^0 - \bar{c}_{v_1/0}^0 \right)^T 1_k \\ \sum_{k=1}^3 \lambda_k \left(\bar{r}_{e/0}^0 - \bar{c}_{v_2/0}^0 \right)^T 1_k \\ \sum_{k=1}^3 \lambda_k \left(\bar{r}_{e/0}^0 - \bar{c}_{v_3/0}^0 \right)^T 1_k \end{bmatrix} \quad (5.69)$$

Debido a que la ecuación (5.37) está reducida, es muy simple expandir y factorizar el término a la extrema derecha de la ecuación anterior como sigue:

$$2 \begin{bmatrix} \sum_{k=1}^3 \lambda_k \left(\bar{r}_{e/0}^0 - \bar{c}_{v_1/0}^0 \right)^T 1_k \\ \sum_{k=1}^3 \lambda_k \left(\bar{r}_{e/0}^0 - \bar{c}_{v_2/0}^0 \right)^T 1_k \\ \sum_{k=1}^3 \lambda_k \left(\bar{r}_{e/0}^0 - \bar{c}_{v_3/0}^0 \right)^T 1_k \end{bmatrix} = 2 \begin{bmatrix} \lambda_1 \left(\bar{r}_{e/0}^0 - \bar{c}_{v_1/0}^0 \right)^T 1_1 + \lambda_2 \left(\bar{r}_{e/0}^0 - \bar{c}_{v_1/0}^0 \right)^T 1_2 + \lambda_3 \left(\bar{r}_{e/0}^0 - \bar{c}_{v_1/0}^0 \right)^T 1_3 \\ \lambda_1 \left(\bar{r}_{e/0}^0 - \bar{c}_{v_2/0}^0 \right)^T 1_1 + \lambda_2 \left(\bar{r}_{e/0}^0 - \bar{c}_{v_2/0}^0 \right)^T 1_2 + \lambda_3 \left(\bar{r}_{e/0}^0 - \bar{c}_{v_2/0}^0 \right)^T 1_3 \\ \lambda_1 \left(\bar{r}_{e/0}^0 - \bar{c}_{v_3/0}^0 \right)^T 1_1 + \lambda_2 \left(\bar{r}_{e/0}^0 - \bar{c}_{v_3/0}^0 \right)^T 1_2 + \lambda_3 \left(\bar{r}_{e/0}^0 - \bar{c}_{v_3/0}^0 \right)^T 1_3 \end{bmatrix}$$

$$2 \begin{bmatrix} \sum_{k=1}^3 \lambda_k \left(\bar{r}_{e/0}^0 - \bar{c}_{v_1/0}^0 \right)^T 1_k \\ \sum_{k=1}^3 \lambda_k \left(\bar{r}_{e/0}^0 - \bar{c}_{v_2/0}^0 \right)^T 1_k \\ \sum_{k=1}^3 \lambda_k \left(\bar{r}_{e/0}^0 - \bar{c}_{v_3/0}^0 \right)^T 1_k \end{bmatrix} = 2 \begin{bmatrix} \left(\bar{r}_{e/0}^0 - \bar{c}_{v_1/0}^0 \right)^T 1_1 & \left(\bar{r}_{e/0}^0 - \bar{c}_{v_1/0}^0 \right)^T 1_2 & \left(\bar{r}_{e/0}^0 - \bar{c}_{v_1/0}^0 \right)^T 1_3 \\ \left(\bar{r}_{e/0}^0 - \bar{c}_{v_2/0}^0 \right)^T 1_1 & \left(\bar{r}_{e/0}^0 - \bar{c}_{v_2/0}^0 \right)^T 1_2 & \left(\bar{r}_{e/0}^0 - \bar{c}_{v_2/0}^0 \right)^T 1_3 \\ \left(\bar{r}_{e/0}^0 - \bar{c}_{v_3/0}^0 \right)^T 1_1 & \left(\bar{r}_{e/0}^0 - \bar{c}_{v_3/0}^0 \right)^T 1_2 & \left(\bar{r}_{e/0}^0 - \bar{c}_{v_3/0}^0 \right)^T 1_3 \end{bmatrix} \underbrace{\begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \end{bmatrix}}_{\bar{\lambda}}. \quad (5.70)$$

Esto permite calcular los valores de $\bar{\lambda} \in \mathbb{R}^{3 \times 1}$ sustituyendo el resultado anterior en la ecuación (5.69)

$$\bar{\lambda} = \underbrace{2 \begin{bmatrix} \left(\begin{smallmatrix} \bar{r}_{e/0}^0 - \bar{c}_{v_1/0}^0 \\ \bar{r}_{e/0}^0 - \bar{c}_{v_2/0}^0 \\ \bar{r}_{e/0}^0 - \bar{c}_{v_3/0}^0 \end{smallmatrix} \right)^T 1_1 & \left(\begin{smallmatrix} \bar{r}_{e/0}^0 - \bar{c}_{v_1/0}^0 \\ \bar{r}_{e/0}^0 - \bar{c}_{v_2/0}^0 \\ \bar{r}_{e/0}^0 - \bar{c}_{v_3/0}^0 \end{smallmatrix} \right)^T 1_2 & \left(\begin{smallmatrix} \bar{r}_{e/0}^0 - \bar{c}_{v_1/0}^0 \\ \bar{r}_{e/0}^0 - \bar{c}_{v_2/0}^0 \\ \bar{r}_{e/0}^0 - \bar{c}_{v_3/0}^0 \end{smallmatrix} \right)^T 1_3 \end{bmatrix}}_{\text{Matriz } A \in \mathbb{R}^{3 \times 3}} \underbrace{\left[(m_e + 3m_v) \ddot{r}_{e/0}^0 - (m_e + 3m_v) g - f \right]}_{\text{Vector } b \in \mathbb{R}^{3 \times 1}} \quad (5.71)$$

$$\bar{\lambda} = \left(2 \begin{bmatrix} \left(\begin{smallmatrix} \bar{r}_{e/0}^0 - \bar{c}_{v_1/0}^0 \\ \bar{r}_{e/0}^0 - \bar{c}_{v_2/0}^0 \\ \bar{r}_{e/0}^0 - \bar{c}_{v_3/0}^0 \end{smallmatrix} \right)^T 1_1 & \left(\begin{smallmatrix} \bar{r}_{e/0}^0 - \bar{c}_{v_1/0}^0 \\ \bar{r}_{e/0}^0 - \bar{c}_{v_2/0}^0 \\ \bar{r}_{e/0}^0 - \bar{c}_{v_3/0}^0 \end{smallmatrix} \right)^T 1_2 & \left(\begin{smallmatrix} \bar{r}_{e/0}^0 - \bar{c}_{v_1/0}^0 \\ \bar{r}_{e/0}^0 - \bar{c}_{v_2/0}^0 \\ \bar{r}_{e/0}^0 - \bar{c}_{v_3/0}^0 \end{smallmatrix} \right)^T 1_3 \end{bmatrix} \right)^{-1} \left[(m_e + 3m_v) \ddot{r}_{e/0}^0 - (m_e + 3m_v) g - f \right]. \quad (5.71)$$

Debido a que los valores de $\bar{\lambda}$ dependen de variables que cambian conforme el tiempo evoluciona, deben calcularse numéricamente **y en tiempo real** (tal como se ha hecho en todos los algoritmos anteriores), mientras se sustituyen en la ecuación (5.58), lo que genera un modelo dinámico cerrado, que realimenta valores entre sí para evitar singularidades. Su algoritmo se muestra a continuación, mientras que su código en MATLAB se encuentra en el apéndice A.16; además, al igual que ocurrió con el algoritmo 13, las masas de los cuerpos rígidos se envían, como parámetro de entrada, a través de la variable m .

Algoritmo 14: Coeficientes $\bar{\lambda}$

```

Entrada:  $(\bar{r}_{e/0}^0, \bar{\theta}, \dot{\bar{\theta}}, \ddot{\bar{\theta}}, a, l, p, m, g, f)$ ; /* Parámetros geométricos */
 $\phi \leftarrow \frac{2}{3}\pi$ ; /* Separación entre articulaciones */
 $J^I(\bar{n}, \bar{r}) \leftarrow$  ecuación (3.26); /* Matriz jacobiana geométrica */
 $\dot{J}^I(\bar{n}, \bar{r}) \leftarrow$  ecuación (4.13); /* Derivada de la matriz jacobiana geométrica */
 $\ddot{r}_{e/0}^0 \leftarrow$  ecuación (4.2); /* Aceleración en el efector final */
 $b \leftarrow (m_e + 3m_v) \ddot{r}_{e/0}^0 - (m_e + 3m_v) g - f$ ; /* Vector b */
for  $i \leftarrow 1$  to 3 do
     $\bar{c}_{v_i/0}^0 \leftarrow$  ecuación (2.11); /* Posición del  $i$ -ésimo centro de rotación */
     $A(i,:) \leftarrow 2 \left[ \left( \bar{r}_{e/0}^0 - \bar{c}_{v_i/0}^0 \right)^T [1 \ 0 \ 0]^T \quad \left( \bar{r}_{e/0}^0 - \bar{c}_{v_i/0}^0 \right)^T [0 \ 1 \ 0]^T \quad \left( \bar{r}_{e/0}^0 - \bar{c}_{v_i/0}^0 \right)^T [0 \ 0 \ 1]^T \right]$ ; /*  $i$ -ésima fila de A */
end
 $\bar{\lambda} \leftarrow A^{-1}b$ 
Salida:  $\bar{\lambda}$ 

```

Este algoritmo requiere de conocer las posiciones, velocidades y aceleraciones de las articulaciones para poder calcular la aceleración **instantánea** del efector final. En un modelo con MATLAB y Simulink, estas se obtienen directamente de los bloques integradores, pero en un robot **real**, se recomienda que los motores utilizados tengan, al menos, un sensor de velocidad para garantizar una correcta **estimación** de su aceleración a través del cálculo numérico presentado en el algoritmo 7. Si solo se puede medir la posición del motor, se deberían utilizar los algoritmos 5 y 7, lo que podría generar errores si el tamaño de paso en ambas derivadas no se escoge adecuadamente.

5.5. Simulación Numérica

Una ventaja de crear algoritmos numéricos es que se pueden utilizar en Simulink, lo que reduce la complejidad del modelo ya que se evita conectar múltiples bloques para cada una de las ecuaciones.

En vez de eso, se crea un único bloque que llama a cada función y cuyos resultados **numéricos** se envían a otros bloques, creando así una simulación completamente funcional. Para este robot, se utilizaron bloques llamados *MATLAB Function*, que permiten llamar, desde Simulink, a todas las funciones que se han desarrollado a lo largo de este documento. Esto se ejemplifica en la siguiente figura

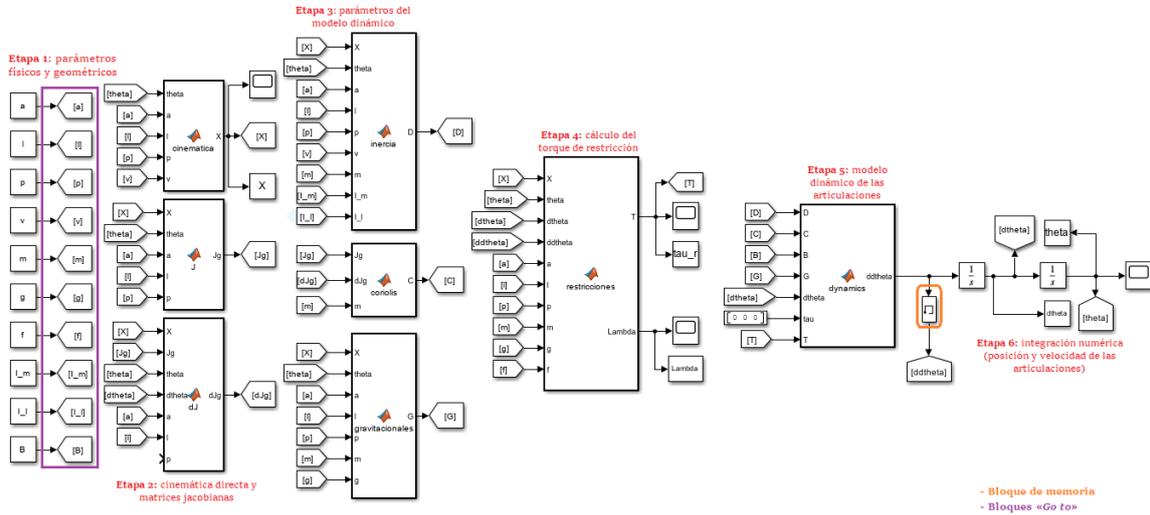


Figura 5.8: Diagrama de Bloques del Modelo Dinámico.

Dentro de los bloques de cada etapa, cuyos códigos se encuentran en el apéndice A.17, solo se debe llamar a la función o funciones implicadas en el proceso que se desea ejecutar. Por ejemplo, en la **Etapa 4** se calculan los coeficientes $\bar{\lambda}$ y el torque de restricción τ_r de la siguiente manera

```

1 function [T, Lambda] = restricciones(X, theta, dtheta, ddtheta, a, l, p, m, g, f)
2
3 % Calcula la matriz de efectos centrífugos y de Coriolis
4 Lambda = lambda_i(X, theta, dtheta, ddtheta, a, l, p, m, g, f);
5
6 % Calcula las restricciones cinemáticas
7 T = constraint(X, theta, a, l, p, Lambda);

```

Figura 5.9: Cálculo de $\bar{\lambda}$ y τ_r en Simulink.

En este caso particular se deben calcular los coeficientes $\bar{\lambda}$ con la función `lambda_i.m` (algoritmo 14 y apéndice A.16), cuyos resultados son enviados a la función `constraint.m` (algoritmo 13 y apéndice A.15), lo que calcula los valores de τ_r , que son enviados como parámetro de salida del bloque y que se requieren en la **Etapa 5** del modelo. En esta existe un *bloque de memoria*, el cual se utiliza para garantizar que, desde el primer ciclo de cálculo, exista un valor de aceleración en las articulaciones θ , así no se crea ningún *bucle algebraico*. Para verificar el funcionamiento del modelo en una simulación, se utilizan los parámetros definidos en la tabla 1, además de los siguientes:

Parámetro	Valor Numérico
θ_0	$[0.7 \ 0.7 \ 0.7]^T$ [rad]
$\dot{\theta}_0$	$[0 \ 0 \ 0]^T$ $\left[\frac{\text{rad}}{\text{seg}}\right]$
m	$[0.4417 \ 0.3039 \ 0.6965]$ [kg]
g	$[0 \ 0 \ -9.80665]^T$ $\left[\frac{m}{s^2}\right]$
f	$[0 \ 0 \ 0]^T$ [Newtons]
τ	$[0 \ 0 \ 0]^T$ [N · m]
I_m	$\begin{bmatrix} 0.1320 & 0_{1 \times 2} \\ 0_{2 \times 1} & 0_{2 \times 2} \end{bmatrix}$ $[kg \cdot m^2]$
I_l	$\begin{bmatrix} 0.9421 & 0.5080 & 0.6982 \\ 0.5080 & 0.2348 & 0.1843 \\ 0.6982 & 0.1843 & 0.0430 \end{bmatrix}$ $[kg \cdot m^2]$
B	$0.75 \cdot \mathbb{I}_{3 \times 3}$ $\left[\frac{kg \cdot m^2}{s}\right]$

Tabla 2: Parámetros del Modelo Dinámico

Estos pueden crearse desde el *command window* en MATLAB o directamente en Simulink, aunque la primera opción da más libertad para modificar los datos. Estableciendo un tiempo de simulación $t = 15$, el modelo arroja los siguientes resultados

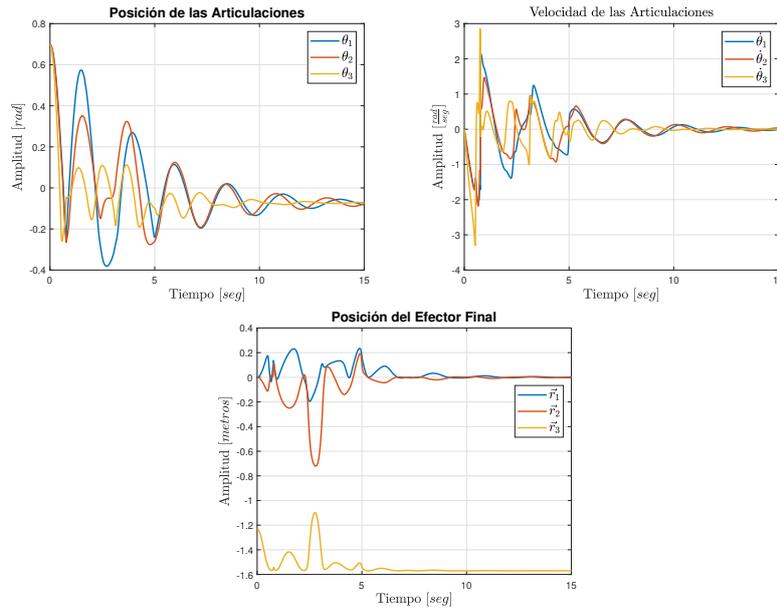


Figura 5.10: Comportamiento de las Articulaciones y el Efector Final en Simulink

Se observa que el comportamiento del efector final $(\vec{r}_1, \vec{r}_2, \vec{r}_3) = (\vec{r}_x, \vec{r}_y, \vec{r}_z)$ es libre, pero errático, en la primera mitad de la simulación, ya que el vector de torque τ es igual a cero y no genera movimientos en las articulaciones. Este comportamiento cambia ya que los fenómenos físicos involucrados, además del torque τ_r , acotan gradualmente el comportamiento de las articulaciones (figura 5.11), lo que lleva al efector final a una posición de reposo, es decir, al final solo se ve afectado por los efectos

gravitacionales, que ocasionan que caiga pero se mantenga unido al resto de cuerpos rígidos en el sistema.

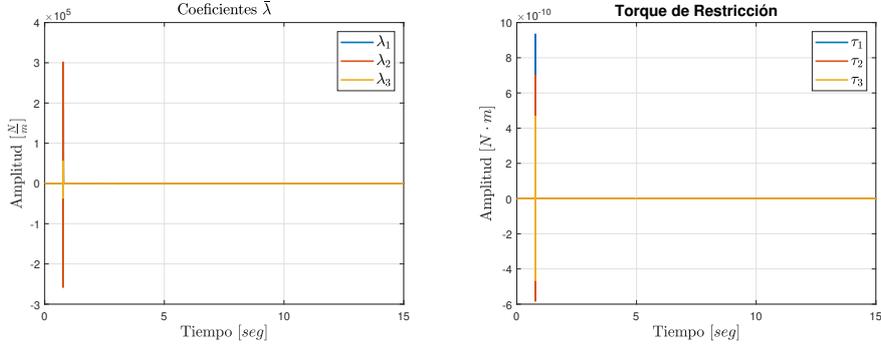


Figura 5.11: Coeficientes $\bar{\lambda}$ y torque de restricción τ_r .

Los resultados permiten concluir que el modelo dinámico funciona correctamente, por lo que se puede proponer que exista una función de control que calcule el torque necesario para que las articulaciones y el efector final tengan un comportamiento definido por el usuario.

6. Sistema de Control

El principio de funcionamiento de un sistema de control es lograr que las coordenadas generalizadas $x \in \mathbb{R}^{n \times 1}$ del robot tengan un comportamiento deseado $x_d \in \mathbb{R}^{n \times 1}$, esto se logra calculando la diferencia entre el comportamiento deseado y el comportamiento medido a lo largo del tiempo, por ejemplo

$$e(t) = x_d - x, \quad (6.1)$$

conforme $t \rightarrow \infty$, el error debe converger o tender a cero para considerar que el sistema de control funciona correctamente.

6.1. Control de Posición y Velocidad de las Articulaciones

Proponer una función de control depende de las coordenadas generalizadas que se están estudiando y del modelo dinámico al que están asociadas. En el caso de las articulaciones, su evolución se describe con el modelo en la ecuación (5.58), el objetivo será encontrar una función de control que permita a las articulaciones alcanzar una posición y velocidad deseadas, para ello se debe expresar el modelo dinámico como una ecuación en espacio de estados tal como se muestra a continuación

$$\begin{aligned} \dot{\theta} &= \dot{\theta} \\ \ddot{\theta} &= D^{-1} [\tau + \tau_r - C\dot{\theta} - B\ddot{\theta} - G] \\ \underbrace{\begin{bmatrix} \dot{\theta} \\ \ddot{\theta} \end{bmatrix}}_{\dot{z}} &= \underbrace{\begin{bmatrix} 0_{3 \times 3} & \mathbb{I}_{3 \times 3} \\ 0_{3 \times 3} & -D^{-1}(C+B) \end{bmatrix}}_A \underbrace{\begin{bmatrix} \theta \\ \dot{\theta} \end{bmatrix}}_z + \underbrace{\begin{bmatrix} 0_{3 \times 1} \\ -D^{-1}G \end{bmatrix}}_{\Phi(t)} + \underbrace{\begin{bmatrix} 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & D^{-1} \end{bmatrix}}_{\mathcal{I}} \underbrace{\begin{bmatrix} 0_{3 \times 1} \\ \tau + \tau_r \end{bmatrix}}_{u_c(t)} \\ \dot{z} &= Az + \Phi(t) + \mathcal{I}u_c(t), \end{aligned} \quad (6.2)$$

donde $z, \dot{z} \in \mathbb{R}^{6 \times 1}$ son el vector de estados y su derivada en función del tiempo; por otro lado, $A \in \mathbb{R}^{6 \times 6}$ es la matriz de estados, $\Phi \in \mathbb{R}^{6 \times 1}$ es el vector de no linealidades, $\mathcal{I} \in \mathbb{R}^{6 \times 6}$ es la matriz de entradas y $u_c \in \mathbb{R}^{6 \times 1}$ es el vector de entradas, del cual se debe encontrar su ecuación para poder controlar el sistema. Para ello se considera la relación entre la posición y velocidad deseadas y medidas, esto es

$$e(t) = \begin{bmatrix} e_p(t) \\ e_v(t) \end{bmatrix} = \begin{bmatrix} \bar{\theta}_d \\ \dot{\bar{\theta}}_d \end{bmatrix} - \begin{bmatrix} \bar{\theta} \\ \dot{\bar{\theta}} \end{bmatrix} \quad (6.3)$$

$$= z_d - z.$$

Ya que no se conocen los valores de z explícitamente, se puede derivar el error en función del tiempo, que para garantizar su estabilidad, en [16], [17] se considera que su evolución se comporta, **idealmente**, como $\dot{e}(t) = -Ke(t)$, por lo tanto

$$\dot{e}(t) = -Ke(t) = -\begin{bmatrix} \dot{\bar{\theta}} \\ \ddot{\bar{\theta}} \end{bmatrix} = -\dot{z} = -Az - \Phi(t) - \mathcal{I}u_c(t), \quad (6.4)$$

donde $K \in \mathbb{R}^{6 \times 6}$ es una matriz de ganancia **constante y definida positiva** para garantizar la estabilidad asintótica del sistema; esta se define, en submatrices, como

$$K = \begin{bmatrix} K_p & K_v \\ K_p & K_v \end{bmatrix}, \quad (6.5)$$

siendo $K_p, K_v \in \mathbb{R}^{3 \times 3}$ las sub matrices de ganancia para la posición y velocidad de las articulaciones, respectivamente. De la ecuación (6.4) solo se debe encontrar u_c , que siguiendo las reglas del álgebra lineal, se llega al siguiente resultado

$$u_c(t) = \mathcal{I}^{-1} [Ke(t) - Az - \Phi(t)]. \quad (6.6)$$

Esto puede incluirse en el modelo en Simulink, sin embargo, vale la pena simplificarlo para reducir el procesamiento en la simulación, o incluso en un sistema real. Primero, es fácil demostrar que la forma inversa de \mathcal{I} es

$$\mathcal{I}^{-1} = \begin{bmatrix} 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & D \end{bmatrix}, \quad (6.7)$$

que si se utiliza con las ecuaciones (6.2), (6.3) y (6.6), se expande y simplifica la función de control de la siguiente manera

$$\begin{aligned} u_c(t) &= \begin{bmatrix} 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & D \end{bmatrix} \left(\begin{bmatrix} K_p & K_v \\ K_p & K_v \end{bmatrix} \begin{bmatrix} e_p(t) \\ e_v(t) \end{bmatrix} - \begin{bmatrix} 0_{3 \times 3} & \mathbb{I}_{3 \times 3} \\ 0_{3 \times 3} & -D^{-1}(C+B) \end{bmatrix} \begin{bmatrix} \bar{\theta} \\ \dot{\bar{\theta}} \end{bmatrix} - \begin{bmatrix} 0_{3 \times 1} \\ -D^{-1}G \end{bmatrix} \right) \\ &= \begin{bmatrix} 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & D \end{bmatrix} \left(\begin{bmatrix} K_p e_p(t) + K_v e_v(t) \\ K_p e_p(t) + K_v e_v(t) \end{bmatrix} - \begin{bmatrix} \dot{\bar{\theta}} \\ -D^{-1}(C+B)\dot{\bar{\theta}} \end{bmatrix} - \begin{bmatrix} 0_{3 \times 1} \\ -D^{-1}G \end{bmatrix} \right) \\ &= \begin{bmatrix} 0_{3 \times 1} \\ D[K_p e_p(t) + K_v e_v(t)] \end{bmatrix} - \begin{bmatrix} 0_{3 \times 1} \\ -DD^{-1}(C+B)\dot{\bar{\theta}} \end{bmatrix} - \begin{bmatrix} 0_{3 \times 1} \\ -DD^{-1}G \end{bmatrix} \\ &= \begin{bmatrix} 0_{3 \times 1} \\ D[K_p e_p(t) + K_v e_v(t)] \end{bmatrix} + \begin{bmatrix} 0_{3 \times 1} \\ (C+B)\dot{\bar{\theta}} \end{bmatrix} + \begin{bmatrix} 0_{3 \times 1} \\ G \end{bmatrix} \\ \begin{bmatrix} 0_{3 \times 1} \\ \tau + \tau_r \end{bmatrix} &= \begin{bmatrix} 0_{3 \times 1} \\ D[K_p e_p(t) + K_v e_v(t)] + (C+B)\dot{\bar{\theta}} + G \end{bmatrix}. \end{aligned} \quad (6.8)$$

Ya que el $u_c(t)$ contiene el torque de las articulaciones τ y el torque de restricción τ_r , se puede definir completamente el torque en las articulaciones como

$$\tau = D [K_p e_p(t) + K_v e_v(t)] + C\dot{\theta} + B\ddot{\theta} + G - \tau_r, \quad (6.9)$$

Este resultado es una función de **Control Proporcional Derivativo con Compensación de Gravedad**, que se puede utilizar con el modelo mostrado en la figura 5.8, agregando dos etapas adicionales:

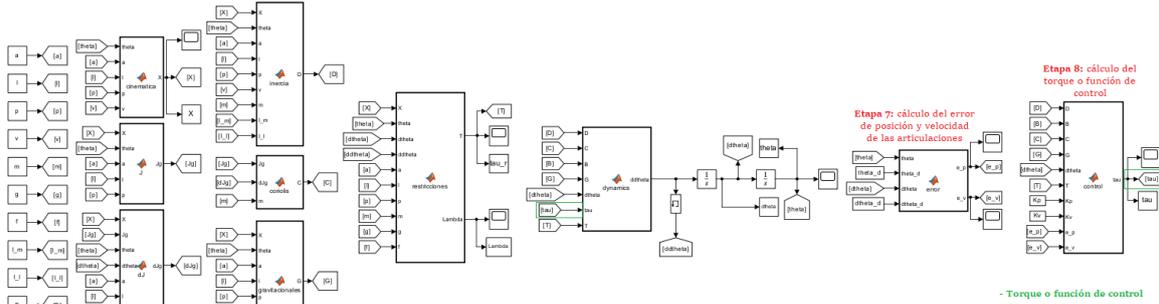


Figura 6.1: Diagrama de Bloques del Sistema de Control.

El código de estos nuevos bloques se encuentra en el apéndice A.18. Para ejemplificar su funcionamiento, se consideran los mismos parámetros definidos en las tablas 1 y 2, además de establecer las condiciones del controlador, las cuales son Las ganancias K_p y K_v se escogen de acuerdo al robot

Parámetro	Valor Numérico
$\bar{\theta}_d$	$[-0.4224 \quad 0.4882 \quad -0.1774]^T$ [rad]
$\dot{\bar{\theta}}_d$	$[0 \quad 0 \quad 0]^T$ $\left[\frac{\text{rad}}{\text{seg}} \right]$
K_p	$10 \cdot \mathbb{I}_{3 \times 3}$
K_v	$10 \cdot \mathbb{I}_{3 \times 3}$

Tabla 3: Parámetros del Sistema de Control de las Articulaciones

y el problema que se quiere resolver, siempre que las matrices sean definidas positivas. Esto lleva al sistema a comportarse tal como se muestra a continuación

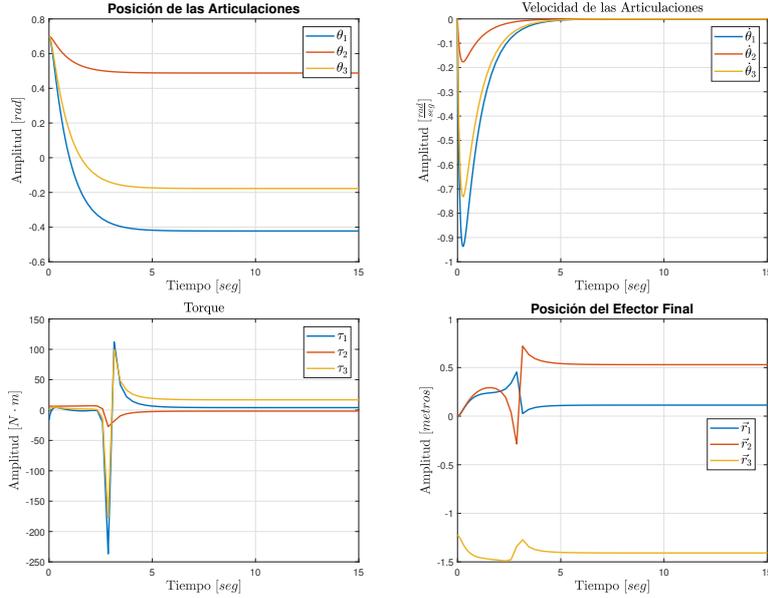


Figura 6.2: Comportamiento Controlado de las Articulaciones y el Efector Final

Se observa que el comportamiento de las articulaciones $\bar{\theta}$ se encuentra acotado y sigue una trayectoria definida gracias al torque τ , que es calculado por la función de control en la **Etapa 8** del modelo en Simulink. Este comportamiento se propaga hacia el efector final, acotando gradualmente su comportamiento en cada uno de sus ejes de desplazamiento $(\vec{r}_1, \vec{r}_2, \vec{r}_3) = (r_x, r_y, r_z)$, llevándolo a una posición específica. Los resultados permiten concluir que el sistema de control funciona correctamente, sin embargo, **este necesita conocer la posición deseada de las articulaciones para alcanzar un punto deseado**, o sea,

1. Se definen la posición y velocidad deseadas en el efector final: $\vec{r}_{e_d/0}^0, \dot{\vec{r}}_{e_d/0}^0$
2. Se calcula la cinemática inversa para conocer la posición deseada en las articulaciones, además de calcular su velocidad: $\bar{\theta}_d, \dot{\bar{\theta}}_d$, algoritmo 3 y ecuación (3.21)
3. Se calcula el torque utilizando el sistema de control: τ , ecuación (6.9)

Esto puede ser lento en aplicaciones que trabajan en tiempo real, sin embargo, no es una limitante, ya que se puede proponer una función de control **cartesiano** que calcule el torque requerido para que el efector final alcance un punto deseado **sin necesariamente calcular la posición de las articulaciones utilizando la cinemática inversa**.

6.2. Control de Posición y Velocidad del Efector Final

A menos que un elemento ajeno al robot controle el efector final, este solo puede actuar bajo los movimientos de las articulaciones, entonces se debe encontrar una relación **dinámica** entre ambos sistemas de coordenadas. Para esto se considera que la aceleración de las articulaciones se puede

calcular utilizando las ecuaciones (3.21) y (4.2) de la siguiente manera

$$\begin{aligned}\ddot{\vec{r}}_{e/0}^0 &= \dot{J}^I(\vec{n}, \vec{r})\dot{\vec{\theta}} + J^I(\vec{n}, \vec{r})\ddot{\vec{\theta}} \\ \ddot{\vec{\theta}} &= J^{-1}\ddot{\vec{r}}_{e/0}^0 - J^{-1}\dot{J}^I\left(J^{-1}\dot{\vec{r}}_{e/0}^0\right).\end{aligned}\quad (6.10)$$

Este resultado se sustituye en el modelo dinámico de la ecuación (5.58), que simplificando los términos, lleva a

$$\begin{aligned}D(\bar{\theta})\left[J^{-1}\ddot{\vec{r}}_{e/0}^0 - J^{-1}\dot{J}^I\left(J^{-1}\dot{\vec{r}}_{e/0}^0\right)\right] + C\left(J^{-1}\dot{\vec{r}}_{e/0}^0\right) + B\dot{\vec{\theta}} + G(\bar{\theta}) &= \tau + \tau_r \\ (DJ^{-1})\ddot{\vec{r}}_{e/0}^0 + \left(CJ^{-1} - DJ^{-1}\dot{J}^I J^{-1}\right)\dot{\vec{r}}_{e/0}^0 + B\dot{\vec{\theta}} + G(\bar{\theta}) &= \tau + \tau_r,\end{aligned}\quad (6.11)$$

en este caso, el producto $B\dot{\vec{\theta}}$ podría expresarse en términos de la velocidad en el efector final, sin embargo, **el rozamiento viscoso planteado en el modelo dinámico contempla estos efectos en las propias articulaciones del sistema, por lo que transformarlos a otro espacio de coordenadas podría ser algo contraproducente, ya que el efector final no solamente se vería afectado por el rozamiento en las articulaciones, sino también por aquél generado por su interacción con el entorno (e.g. el aire).** Para estudiar la posición y velocidad del efector final, se expresa el resultado anterior como una ecuación en espacio de estados:

$$\begin{aligned}\dot{\vec{r}}_{e/0}^0 &= \dot{\vec{r}}_{e/0}^0 \\ \ddot{\vec{r}}_{e/0}^0 &= (DJ^{-1})^{-1}\left[\tau + \tau_r - \left(CJ^{-1} - DJ^{-1}\dot{J}^I J^{-1}\right)\dot{\vec{r}}_{e/0}^0 - B\dot{\vec{\theta}} - G\right] \\ \underbrace{\begin{bmatrix} \dot{\vec{r}}_{e/0}^0 \\ \ddot{\vec{r}}_{e/0}^0 \end{bmatrix}}_{\dot{z}_{\mathcal{X}}} &= \underbrace{\begin{bmatrix} 0_{3 \times 3} & \mathbb{I}_{3 \times 3} \\ 0_{3 \times 3} & -(DJ^{-1})^{-1}CJ^{-1} + \dot{J}^I J^{-1} \end{bmatrix}}_{A_{\mathcal{X}}}\underbrace{\begin{bmatrix} \vec{r}_{e/0}^0 \\ \dot{\vec{r}}_{e/0}^0 \end{bmatrix}}_{z_{\mathcal{X}}} + \underbrace{\begin{bmatrix} 0_{3 \times 1} \\ - (DJ^{-1})^{-1}[B\dot{\vec{\theta}} + G] \end{bmatrix}}_{\Phi_{\mathcal{X}}(t)} + \underbrace{\begin{bmatrix} 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & (DJ^{-1})^{-1} \end{bmatrix}}_{\mathcal{I}_{\mathcal{X}}}\underbrace{\begin{bmatrix} 0_{3 \times 1} \\ \tau + \tau_r \end{bmatrix}}_{u_{c_{\mathcal{X}}}(t)} \\ \dot{z}_{\mathcal{X}} &= A_{\mathcal{X}}z_{\mathcal{X}} + \Phi_{\mathcal{X}}(t) + \mathcal{I}_{\mathcal{X}}u_{c_{\mathcal{X}}}(t),\end{aligned}\quad (6.12)$$

donde $z_{\mathcal{X}}, \dot{z}_{\mathcal{X}} \in \mathbb{R}^{6 \times 1}$ son el vector de estados y su derivada en función del tiempo; por otro lado, $A_{\mathcal{X}} \in \mathbb{R}^{6 \times 6}$ es la matriz de estados, $\Phi_{\mathcal{X}} \in \mathbb{R}^{6 \times 1}$ es el vector de no linealidades, $\mathcal{I}_{\mathcal{X}} \in \mathbb{R}^{6 \times 6}$ es la matriz de entradas y $u_{c_{\mathcal{X}}} \in \mathbb{R}^{6 \times 1}$ es el vector de entradas, del cual se debe encontrar su ecuación para poder controlar el sistema. Para ello se considera la relación entre la posición y velocidad deseadas y medidas, esto es

$$\begin{aligned}e_{\mathcal{X}}(t) &= \begin{bmatrix} e_p(t) \\ e_v(t) \end{bmatrix} = \begin{bmatrix} \vec{r}_{e_d/0}^0 \\ \dot{\vec{r}}_{e_d/0}^0 \end{bmatrix} - \begin{bmatrix} \vec{r}_{e/0}^0 \\ \dot{\vec{r}}_{e/0}^0 \end{bmatrix} \\ &= z_{\mathcal{X}_d} - z_{\mathcal{X}}.\end{aligned}\quad (6.13)$$

Ya que no se conocen los valores de $z_{\mathcal{X}}$ explícitamente, se puede derivar el error en función del tiempo, que para garantizar su estabilidad, en [16], [17] se considera que su evolución se comporta, **idealmente**, como $\dot{e}_{\mathcal{X}}(t) = -Ke_{\mathcal{X}}(t)$, por lo tanto

$$\dot{e}_{\mathcal{X}}(t) = -Ke_{\mathcal{X}}(t) = -\begin{bmatrix} \vec{r}_{e/0}^0 \\ \dot{\vec{r}}_{e/0}^0 \end{bmatrix} = -\dot{z}_{\mathcal{X}} = -A_{\mathcal{X}}z_{\mathcal{X}} - \Phi_{\mathcal{X}}(t) - \mathcal{I}_{\mathcal{X}}u_{c_{\mathcal{X}}}(t),\quad (6.14)$$

donde $K \in \mathbb{R}^{6 \times 6}$ es una matriz de ganancia **constante y definida positiva** para garantizar la estabilidad asintótica del sistema; esta se define, en submatrices, como en la ecuación (6.5). De la

ecuación anterior solo se debe encontrar u_c , que siguiendo las reglas del álgebra lineal, se llega al siguiente resultado

$$u_{c\mathcal{X}}(t) = \mathcal{I}_{\mathcal{X}}^{-1} [K e_{\mathcal{X}}(t) - A_{\mathcal{X}} z_{\mathcal{X}} - \Phi_{\mathcal{X}}(t)]. \quad (6.15)$$

Esto puede incluirse en el modelo en Simulink, sin embargo, vale la pena simplificarlo para reducir el procesamiento en la simulación, o incluso en un sistema real. Primero, es fácil demostrar que la forma inversa de $\mathcal{I}_{\mathcal{X}}$ es

$$\mathcal{I}_{\mathcal{X}}^{-1} = \begin{bmatrix} 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & DJ^{-1} \end{bmatrix}, \quad (6.16)$$

que si se utiliza con las ecuaciones (6.12), (6.13) y (6.15), se expande y simplifica la función de control de la siguiente manera

$$\begin{aligned} u_{c\mathcal{X}}(t) &= \begin{bmatrix} 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & DJ^{-1} \end{bmatrix} \left(\begin{bmatrix} K_p & K_v \\ K_p & K_v \end{bmatrix} \begin{bmatrix} e_p(t) \\ e_v(t) \end{bmatrix} - \begin{bmatrix} 0_{3 \times 3} & \mathbb{I}_{3 \times 3} \\ 0_{3 \times 3} & -(DJ^{-1})^{-1}CJ^{-1} + j^I J^{-1} \end{bmatrix} \begin{bmatrix} \dot{r}_{e/0}^0 \\ \ddot{r}_{e/0}^0 \end{bmatrix} - \begin{bmatrix} 0_{3 \times 1} \\ -(DJ^{-1})^{-1} [B\dot{\theta} + G] \end{bmatrix} \right) \\ &= \begin{bmatrix} 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & DJ^{-1} \end{bmatrix} \left(\begin{bmatrix} K_p e_p(t) + K_v e_v(t) \\ K_p e_p(t) + K_v e_v(t) \end{bmatrix} - \begin{bmatrix} \dot{r}_{e/0}^0 \\ [- (DJ^{-1})^{-1}CJ^{-1} + j^I J^{-1}] \ddot{r}_{e/0}^0 \end{bmatrix} + \begin{bmatrix} 0_{3 \times 1} \\ (DJ^{-1})^{-1} [B\dot{\theta} + G] \end{bmatrix} \right) \\ &= \begin{bmatrix} 0_{3 \times 1} \\ DJ^{-1} [K_p e_p(t) + K_v e_v(t)] \end{bmatrix} - \begin{bmatrix} 0_{3 \times 1} \\ DJ^{-1} [- (DJ^{-1})^{-1}CJ^{-1} + j^I J^{-1}] \ddot{r}_{e/0}^0 \end{bmatrix} + \begin{bmatrix} 0_{3 \times 1} \\ B\dot{\theta} + G \end{bmatrix} \\ \begin{bmatrix} 0_{3 \times 1} \\ \tau + \tau_r \end{bmatrix} &= \begin{bmatrix} 0_{3 \times 1} \\ DJ^{-1} [K_p e_p(t) + K_v e_v(t)] + [CJ^{-1} - DJ^{-1} j^I J^{-1}] \ddot{r}_{e/0}^0 + B\dot{\theta} + G \end{bmatrix}. \end{aligned} \quad (6.17)$$

Ya que el $u_c(t)$ contiene el torque de las articulaciones τ y el torque de restricción τ_r , se puede definir completamente el torque en las articulaciones como

$$\tau = DJ^{-1} [K_p e_p(t) + K_v e_v(t) - j^I J^{-1} \ddot{r}_{e/0}^0] + (CJ^{-1}) \ddot{r}_{e/0}^0 + B\dot{\theta} + G - \tau_r, \quad (6.18)$$

Este resultado es una función de **Control Proporcional Derivativo con Compensación de Gravedad**, que se puede utilizar con el modelo mostrado en la figura 5.8, agregando dos etapas adicionales y un nuevo bloque en la **Etapa 2**:

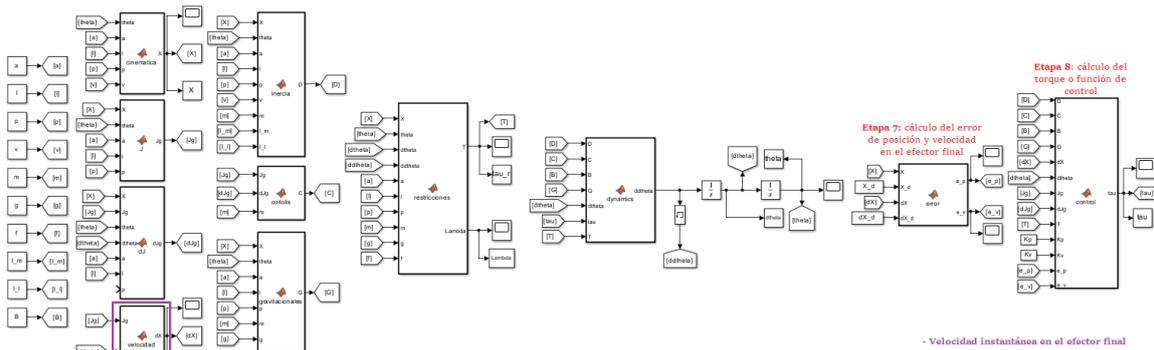


Figura 6.3: Diagrama de Bloques del Sistema de Control Cartesiano.

El código de estos nuevos bloques se encuentra en el apéndice A.19. Para ejemplificar su funcionamiento, se consideran los mismos parámetros definidos en las tablas 1 y 2, además de establecer las condiciones del controlador, las cuales son

Parámetro	Valor Numérico
$\bar{r}_{e_d/0}^0$	$[0.1135 \quad 0.5298 \quad -1.4082]^T$ [metros]
$\dot{\bar{r}}_{e_d/0}^0$	$[0 \quad 0 \quad 0]^T$ [$\frac{m}{s}$]
K_p	$10 \cdot \mathbb{I}_{3 \times 3}$
K_v	$10 \cdot \mathbb{I}_{3 \times 3}$

Tabla 4: Parámetros del Sistema de Control Cartesiano

Las ganancias K_p y K_v se escogen de acuerdo al robot y el problema que se quiere resolver, siempre que las matrices sean definidas positivas. Esto lleva al sistema a comportarse tal como se muestra a continuación

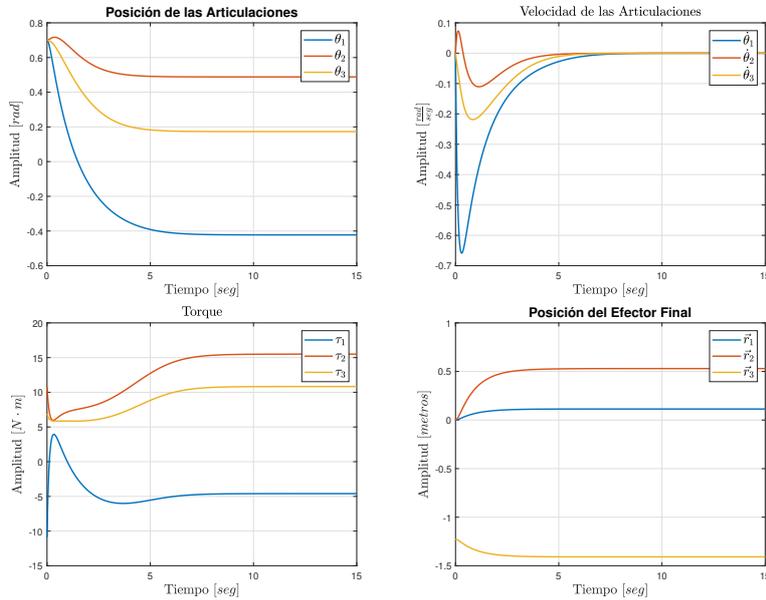


Figura 6.4: Comportamiento Controlado de las Articulaciones y el Efector Final

Se observa que el comportamiento de las articulaciones $\bar{\theta}$ y $\dot{\bar{\theta}}$ se genera automáticamente al suministrar el torque τ , quien es calculado por la función de control en la **Etapa 8** del modelo en Simulink. Este comportamiento se alimenta del cálculo entre la diferencia de la posición y velocidad deseadas y medidas en el efector final, quienes gradualmente llegan al punto deseado, que es el mismo mostrado en la figura 6.2. Si se comparan ambos controladores, el cartesiano calcula un torque sin sobretiros y esto, automáticamente, desplaza las articulaciones a una posición que permite al efector final alcanzar su posición deseada, por lo que el cálculo de la cinemática inversa se omite utilizando este tipo de controlador.

Apéndices

A.1. Cinemática Directa

```
function [x, y, z] = cinematicaDirecta(theta, a, l, p, v)

    phi = (2 / 3) * pi;

    %% Variables para almacenar datos sobre los centros instantáneos de rotación
    c = zeros(3, 3);
    w = zeros(1, 3);

    %% Cálculo del Centro Instantáneo de Rotación
    for i = 1 : 3

        %% Coordenadas
        c(:, i) = [+sin((i - 1) * phi) * (a + (1 * sin(theta(i)))) - p
                  -cos((i - 1) * phi) * (a + (1 * sin(theta(i)))) - p
                  -1 * cos(theta(i))];

        %% Suma de los cuadrados de las coordenadas
        w(:, i) = c(1, i)^2 + c(2, i)^2 + c(3, i)^2;

    end

    %% Vectores unitarios normales a los planos en las esferas
    U = [c(1, 2) - c(1, 1)      % [c_x_2 - c_x_1
        c(2, 2) - c(2, 1)      % c_y_2 - c_y_1
        c(3, 2) - c(3, 1)];    % c_z_2 - c_z_1

    V = [c(1, 3) - c(1, 1)      % [c_x_3 - c_x_1
        c(2, 3) - c(2, 1)      % c_y_3 - c_y_1
        c(3, 3) - c(3, 1)];    % c_z_3 - c_z_1

    %% Dirección del vector perpendicular a los unitarios: d = -u x v
    d = cross(-U, V);

    %% Origen de la recta que interseca las esferas
    v0 = [c(1, 2) - c(1, 1), c(2, 2) - c(2, 1)
          c(1, 3) - c(1, 1), c(2, 3) - c(2, 1)] \ (0.5 * [w(:, 2) - w(:, 1)
                                                           w(:, 3) - w(:, 1)]);

    %% Raíces de la ecuación de la esfera: x^2 + y^2 + z^2 - 2 (c_x_i * x + c_y_i * y + c_z_i * z) + w_i = v_i^2

    %% Cálculo de los valores de z a partir de un polinomio m * z^2 + n * z + p = 0
    m = (d(1) / d(3))^2 + (d(2) / d(3))^2 + 1;
    n = 2 * ((v0(1) - c(1, 1)) * (d(1) / d(3)) + (v0(2) - c(2, 1)) * (d(2) / d(3)) - c(3, 1));
    p = v0(1)^2 + v0(2)^2 - 2 * (c(1, 1) * v0(1) + c(2, 1) * v0(2)) + w(1, 1) - v^2;
    z = min( ...
        real( ...
            roots([m n p]) ...
        ) ...
    );

    %% x(z)
    x = v0(1) + (z * (d(1) / d(3)));

    %% y(z)
    y = v0(2) + (z * (d(2) / d(3)));

end
```

A.2. Matriz Jacobiana Geométrica

```
function Jg = matrizJacobiana(X, theta, a, l, p)

    %% Ángulo de separación entre articulaciones
    phi = (2 / 3) * pi;

    %% Inicializa las matrices jacobianas

    % Crea matrices vacías NUMÉRICAS
    Jx = zeros(3, 3);
    Jtheta = zeros(3, 3);

    %% Itera sobre cada articulación
    for i = 1 : 3

        % Cálculo de los términos de la matriz
        axis = [-cos((i - 1) * phi) -sin((i - 1) * phi) 0]';
        r_pi_i = [X(1) - sin((i - 1) * phi) * (a - p + l * sin(theta(i)))
                  X(2) + cos((i - 1) * phi) * (a - p + l * sin(theta(i)))
                  X(3) + l * cos(theta(i))];
        r_i_ai = [+l * sin((i - 1) * phi) * sin(theta(i))
                  -l * cos((i - 1) * phi) * sin(theta(i))
                  -l * cos(theta(i))];

        % Matriz jacobiana del efector final
        Jx(i, :) = r_pi_i';

        % Matriz jacobiana de las articulaciones
        Jtheta(i, i) = axis' * cross(r_i_ai, r_pi_i);

    end

    Jg = Jx \ Jtheta;

end
```

A.3. Cinemática Inversa

```
function [theta] = cinematicaInversa(Xd, K, theta0, a, l, p, v, dt)

    %% Condiciones iniciales
    theta = theta0;

    %% Iteraciones para delimitar el tiempo de cálculo
    for k = 1 : 1500

        %% Cinemática directa
        [x, y, z] = cinematicaDirecta(theta(:, k), a, l, p, v);

        %% Cálculo del error
        error = Xd - [x y z]';

        %% Si el máximo error es menor a una milésima
        if max(abs(error)) <= 1e-3

            %% La solución fue encontrada y el ciclo iterativo termina
            break;

        %% Si lo anterior no se cumple,
        else

            %% Se calcula la matriz jacobiana del robot
            J = matrizJacobiana([x y z]', theta(:, k), a, l, p);

            %% Integral numérica de la ecuación
            theta(:, k + 1) = theta(:, k) + (J \ (K * error)) * dt;

        end

    end

end

end

end
```

A.4. Derivada de una Matriz Jacobiana Geométrica

```

function dJg = derivadaMatrizJacobiana(X, Jg, theta, dtheta, a, l, p)

    %% Ángulo de separación entre articulaciones
    phi = (2 / 3) * pi;

    %% Inicializa las matrices jacobianas

    % Crea matrices vacías NUMÉRICAS
    Jx = zeros(3, 3);
    dJtheta = zeros(3, 3);

    %% Itera sobre cada articulación
    for i = 1 : 3

        %% Cálculo de los términos en la matriz
        axis = [-cos((i - 1) * phi) -sin((i - 1) * phi) 0]';

        % Posición y Velocidad de r_pi_i
        r_pi_i = [X(1) - sin((i - 1) * phi) * (a - p + l * sin(theta(i)))
                  X(2) + cos((i - 1) * phi) * (a - p + l * sin(theta(i)))
                  X(3) + l * cos(theta(i))];
        dot_r_pi_i = (Jg * dtheta) - ([+1 * sin((i - 1) * phi) * cos(theta(i))
                                       -1 * cos((i - 1) * phi) * cos(theta(i))
                                       +1 * sin(theta(i))] * dtheta(i));

        % Posición y Velocidad de r_i_ai
        r_i_ai = [+1 * sin((i - 1) * phi) * sin(theta(i))
                  -1 * cos((i - 1) * phi) * sin(theta(i))
                  -1 * cos(theta(i))];
        dot_r_i_ai = [+1 * sin((i - 1) * phi) * cos(theta(i))
                      -1 * cos((i - 1) * phi) * cos(theta(i))
                      +1 * sin(theta(i))] * dtheta(i);

        %% Matriz jacobiana del efector final
        Jx(i, :) = r_pi_i';

        %% Parte de la matriz jacobiana de las articulaciones
        dJtheta(i, i) = axis' * (cross(dot_r_i_ai, r_pi_i) + ...
                                cross(r_i_ai, dot_r_pi_i + Jg * dtheta));

    end

    %% Derivada de la matriz jacobiana geométrica
    dJg = Jx \ (dJtheta - [(Jg * dtheta)' * Jg; (Jg * dtheta)' * Jg; (Jg * dtheta)' * Jg]);

end

```

A.5. Velocidad Instantánea de las Articulaciones

```
function [dot_theta] = velocidadArticulaciones(theta, tmax)

    %% Obtiene el numero de elementos calculados
    [m, n] = size(theta);

    %% Calcula el tamaño de paso para la derivada numerica
    dt = tmax / n;

    %% Inicializa la velocidad con ceros
    dot_theta = zeros(m, n);

    %% Itera sobre cada elemento calculado
    for k = 1 : n - 1

        %% Calcula la velocidad de las articulaciones de forma numerica
        dot_theta(:, k) = (theta(:, k + 1) - theta(:, k)) / dt;

    end

end
```

A.6. Velocidad Instantánea del Efector Final

```
function [dot_x] = velocidadEfector(theta, dot_theta, a, l, p, v)

    %% Obtiene el numero de elementos calculados
    [~, n] = size(dot_theta);

    %% Inicializa la velocidad con ceros
    dot_x = zeros(3, n);

    %% Itera sobre cada elemento calculado
    for k = 1 : n

        %% Calcula la cinemática directa
        [x, y, z] = cinematicaDirecta(theta(:, k), a, l, p, v);

        %% Calcula la matriz jacobiana del elemento actual
        J = matrizJacobiana([x y z]', theta(:, k), a, l, p);

        %% Calcula la velocidad
        dot_x(:, k) = J * dot_theta(:, k);

    end

end
```

A.7. Aceleración Instantánea de las Articulaciones

```
function [ddot_theta] = aceleracionArticulaciones(dot_theta, tmax)

    %% Obtienefuncion el numero de elementos calculados
    [m, n] = size(dot_theta);

    %% Calcula el tamaño de paso para la derivada numerica
    dt = tmax / n;

    %% Inicializa la velocidad con ceros
    ddot_theta = zeros(m, n);

    %% Itera sobre cada elemento calculado
    for k = 1 : n - 1

        %% Calcula la velocidad de las articulaciones de forma numerica
        ddot_theta(:, k) = (dot_theta(:, k + 1) - dot_theta(:, k)) / dt;

    end

end
```

A.8. Aceleración Instantánea del Efector Final

```
function [ddot_x] = aceleracionEfector(theta, dot_theta, ddot_theta, a, l, p, v)

    %% Obtiene el numero de elementos calculados
    [~, n] = size(ddot_theta);

    %% Inicializa la velocidad con ceros
    ddot_x = zeros(3, n);

    %% Itera sobre cada elemento calculado
    for k = 1 : n

        %% Cinemática directa
        [x, y, z] = cinematicaDirecta(theta(:, k), a, l, p, v);

        %% Calcula la matriz jacobiana del elemento actual
        J = matrizJacobiana([x y z]', theta(:, k), a, l, p);

        %% Calcula la derivada de la matriz jacobiana del elemento actual
        dJ = derivadaMatrizJacobiana([x y z]', J, theta(:, k), dot_theta(:, k), a, l, p);

        %% Calcula la aceleración
        ddot_x(:, k) = (dJ * dot_theta(:, k)) + (J * ddot_theta(:, k));

    end

end
```

A.9. Matrices Jacobianas Geométricas del Cuerpo Rígido l_i

```
function [Jv_l_i, Jw_l_i] = matrizJacobiana_l_i(theta, l_i)

    %% Ángulo de separación entre articulaciones
    phi = (2 / 3) * pi;

    %% Celdas para almacenar las matrices jacobianas
    Jv_l_i = cell(3, 1);
    Jw_l_i = cell(3, 1);

    %% Itera sobre cada articulación
    for i = 1 : 3

        %% Crea matrices vacías NUMÉRICAS
        Jv = zeros(3, 3);
        Jw = zeros(3, 3);

        %% Cálculo de los términos de la matriz
        axis = [-cos((i - 1) * phi) -sin((i - 1) * phi) 0]';
        r_i_ai = [+l_i * sin((i - 1) * phi) * sin(theta(i))
                  -l_i * cos((i - 1) * phi) * sin(theta(i))
                  -l_i * cos(theta(i))];

        %% Matriz jacobiana de velocidad lineal
        Jv(:, i) = cross(axis, r_i_ai);
        Jv_l_i{i} = Jv;

        %% Matriz jacobiana de velocidad angular
        Jw(:, i) = axis;
        Jw_l_i{i} = Jw;

    end

end
```

A.10. Matriz de Rotación del Cuerpo Rígido l_i

```
function R = matrizRotacion_li(theta)

    %% Ángulo de separación entre articulaciones
    phi = (2 / 3) * pi;

    %% Inicializa las celdas para almacenar las matrices de rotación
    R = cell(3, 1);

    %% Itera sobre cada articulación
    for i = 1 : 3

        %% Calcula y almacena la matriz de rotación de  $l_i$ 
        R{i} = [cos((i - 1) * phi), -sin((i - 1) * phi) * cos(theta(i)), -sin((i - 1) * phi) * sin(theta(i))
                sin((i - 1) * phi),  cos((i - 1) * phi) * cos(theta(i)),  cos((i - 1) * phi) * sin(theta(i))
                0,                    - sin(theta(i)),                    cos(theta(i))];

    end

end
```

A.11. Tensor de Inercia respecto al Marco Inercial

```
function I_0 = transformacionInercia(I, R)

    %% Realiza la transformación con la matriz de rotación
    I_0 = R' * I * R;

end
```

A.12. Matriz de Inercia del Robot

```
function D = matrizInercia(X, theta, a, l, p, m, I)

    %% Matriz jacobiana geométrica
    Jg = matrizJacobiana(X, theta, a, l, p);

    %% Matrices jacobianas de los cuerpos l_i
    [Jv_l_i, Jw_l_i] = matrizJacobiana_l_i(theta, l);

    %% Matrices de rotación de los cuerpos rígidos l_i
    R_li = matrizRotacion_li(theta);

    %% Inicializa la matriz de inercia
    D = (m(3) + 3 * m(2)) * (Jg' * Jg);

    %% Itera sobre cada articulación
    for j = 1 : 3

        %% Transformación de los tensores de inercia
        I_m_j_0 = transformacionInercia(I{1}, R_li{j});
        I_l_j_0 = transformacionInercia(I{2}, R_li{j});

        %% i-ésima matriz de inercia
        D = D + (m(1) + m(2)) * (Jv_l_i{j}' * Jv_l_i{j}) ...
            + (Jw_l_i{j}' * (I_m_j_0 + I_l_j_0) * Jw_l_i{j});

    end

end
```

A.13. Matriz de Efectos Centrífugos y de Coriolis

```
function C = coriolis(Jg, dJg, m)

    %% Calcula la matriz
    C = (m(3) + 3 * m(2)) * ((dJg' * Jg) + (Jg' * dJg));

end
```

A.14. Matriz de Efectos Gravitacionales

```
function G = matrizGravitacionales(X, theta, a, l, p, m, g)

    %% Matriz jacobiana geométrica
    Jg = matrizJacobiana(X, theta, a, l, p);

    %% Matrices jacobianas de los cuerpos l_i
    [Jv_l_i, ~] = matrizJacobiana_l_i(theta, l);

    %% Inicializa los efectos gravitacionales
    G = (m(3) + 3 * m(2)) * Jg';

    %% Itera sobre cada articulación
    for j = 1 : 3

        %% Calcula los efectos gravitacionales de la j-ésima articulación
        G = G + m(1) * Jv_l_i{j}' * g;

    end

    %% Multiplica por la aceleración gravitacional (y garantiza la dirección de los cuerpos rígidos)
    G = -G * g;

end
```

A.15. Torque de Restricción τ_r

```
function T = constraint(X, theta, a, l, p, Lambda)

    %% Ángulo de separación entre articulaciones
    phi = (2 / 3) * pi;

    %% Matriz jacobiana geométrica
    Jg = matrizJacobiana(X, theta, a, l, p);

    %% Inicializa el torque
    T = zeros(3, 1);

    %% Itera sobre los centros instantáneos de rotación
    for i = 1 : 3

        %% Posición del i-ésimo centro instantáneo de rotación
        C_i = [ sin((i - 1) * phi) * (a - p + l * sin(theta(i)))
              -cos((i - 1) * phi) * (a - p + l * sin(theta(i)))
              -1 * cos(theta(i))];

        %% Derivada de la posición del centro de rotación
        Rn_x_r_i_ai = [ l * sin((i - 1) * phi) * cos(theta(i))
                      -l * cos((i - 1) * phi) * cos(theta(i))
                      l * sin(theta(i))];

        %% Itera sobre cada lambda_k (dos ciclos iterativos anidados hacen al código más lento)
        for k = 1 : 3

            %% Calcula la celda (i, k) de la restricción
            if i == k

                %% La derivada (d theta_i / d theta_k) = 1
                T(i, :) = T(i, :) + 2 * (Lambda(k) * (X - C_i)' * (Jg(:, k) - Rn_x_r_i_ai));

            else

                %% La derivada (d theta_i / d theta_k) = 0
                T(i, :) = T(i, :) + 2 * (Lambda(k) * (X - C_i)' * Jg(:, k));

            end

        end

    end

end

end

end
```

A.16. Coeficientes $\bar{\lambda}$

```
function Lambda = lambda_i(X, theta, dtheta, ddtheta, a, l, p, m, g, f)

    %% Ángulo de separación entre articulaciones
    phi = (2 / 3) * pi;

    %% Matrices jacobianas
    Jg = matrizJacobiana(X, theta, a, l, p);
    dJg = derivadaMatrizJacobiana(X, Jg, theta, dtheta, a, l, p);

    %% Aceleración del efector final
    ddX = (dJg * dtheta) + (Jg * ddtheta);

    %% Términos del modelo dinámico cartesiano
    b = (m(3) + 3 * m(2)) * ddX - (m(3) + 3 * m(2)) * g - f;

    %% Inicializa la matriz que multiplica a Lambda
    A = zeros(3,3);

    %% Itera sobre los centros instantáneos de rotación
    for i = 1 : 3

        %% Posición del i-ésimo centro instantáneo de rotación
        C_i = [ sin((i - 1) * phi) * (a - p + l * sin(theta(i)))
                -cos((i - 1) * phi) * (a - p + l * sin(theta(i)))
                -1 * cos(theta(i))];

        %% i-ésima fila de la matriz «A» para la ecuación
        A(i, :) = 2 * [(X - C_i)' * [1 0 0]', (X - C_i)' * [0 1 0]', (X - C_i)' * [0 0 1]'];

    end

    %% Calcula los valores de lambda
    Lambda = A \ b;

end
```

A.17. Bloques del Modelo Dinámico en Simulink

■ Etapa 2

```
function X = cinematica(theta, a, l, p, v)

% Calcula la cinemática directa
[x, y, z] = cinematicaDirecta(theta, a, l, p, v);

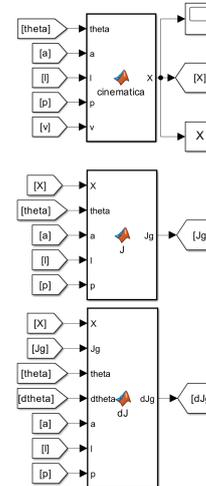
% Envía los resultados en un solo vector
X = [x; y; z];

function Jg = J(X, theta, a, l, p)

% Calcula la matriz jacobiana geométrica
Jg = matrizJacobiana(X, theta, a, l, p);

function dJg = dJ(X, Jg, theta, dtheta, a, l, p)

% Calcula la derivada de la matriz jacobiana geométrica
dJg = derivadaMatrizJacobiana(X, Jg, theta, dtheta, a, l, p);
```



■ Etapa 3

```
function D = inercia(X, theta, a, l, p, v, m, I_m, I_l)

% Agrupa los tensores de inercia en una variable tipo celda
I = {I_m, I_l};

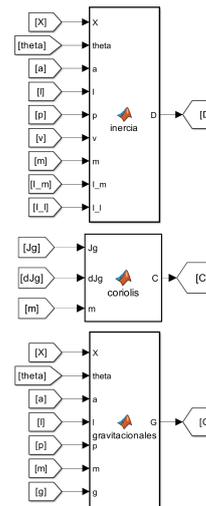
% Calcula la matriz de inercia
D = matrizInercia(X, theta, a, l, p, m, I);

function C = coriolis(Jg, dJg, m)

% Calcula la matriz de efectos centrífugos y de Coriolis
C = coriolis(Jg, dJg, m);

function G = gravitacionales(X, theta, a, l, p, m, g)

% Calcula la matriz de efectos gravitacionales
G = matrizGravitacionales(X, theta, a, l, p, m, g);
```

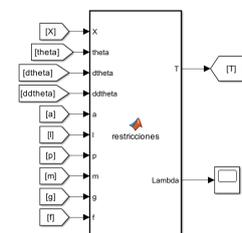


■ Etapa 4

```
function [T, Lambda] = restricciones(X, theta, dtheta, ddtheta, a, l, p, m, g, f)

% Calcula la matriz de efectos centrífugos y de Coriolis
Lambda = lambda_i(X, theta, dtheta, ddtheta, a, l, p, m, g, f);

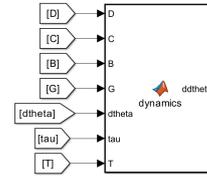
% Calcula las restricciones cinemáticas
T = constraint(X, theta, a, l, p, Lambda);
```



■ **Etapa 5**

```
function ddtheta = dynamics(D, C, B, G, dtheta, tau, T)

% Calcula el modelo dinámico
ddtheta = D \ (tau + T - (C + B) * dtheta - G);
```

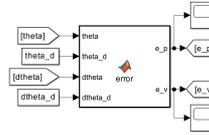


A.18. Bloques del Sistema de Control de las Articulaciones en Simulink

■ Etapa 7

```
function [e_p, e_v] = error(theta, theta_d, dtheta, dtheta_d)

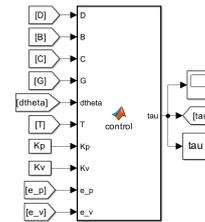
% Error del sistema de control
e_p = theta_d - theta;
e_v = dtheta_d - dtheta;
```



■ Etapa 8

```
function tau = control(D, B, C, G, dtheta, T, Kp, Kv, e_p, e_v)

% Torque
tau = D * (Kp * e_p + Kv * e_v) + (C + B) * dtheta + G - T;
```

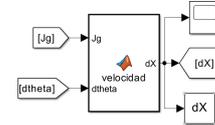


A.19. Bloques del Sistema de Control Cartesiano en Simulink

■ Etapa 2

```
function dX = velocidad(Jg, dtheta)

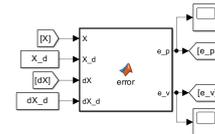
    % Calcula la velocidad en el efector final
    dX = Jg * dtheta;
```



■ Etapa 7

```
function [e_p, e_v] = error(X, X_d, dX, dX_d)

    % Error del sistema de control
    e_p = X_d - X;
    e_v = dX_d - dX;
```

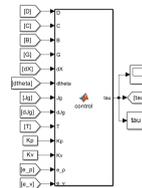


■ Etapa 8

```
function tau = control(D, C, B, G, dX, dtheta, Jg, dJg, T, Kp, Kv, e_p, e_v)

    % Forma inversa de la matriz jacobiana geométrica
    Jinv = Jg \ eye(3);

    % Torque
    tau = (D * Jinv) * ((Kp * e_p) + (Kv * e_v) - (dJg * Jinv * dX)) ...
        + (C * Jinv) * dX + (B * dtheta) + G - T;
```



Referencias

- [1] J. Merlet, *Parallel Robots* (Solid Mechanics and Its Applications). Springer Netherlands, 2005, ISBN: 9781402041327.
- [2] A. Khorasani, S. Gholami y H. Taghirad, «Optimization of KNTU Delta Robot for Pick and Place Application,» oct. de 2015. DOI: [10.1109/ICRoM.2015.7367772](https://doi.org/10.1109/ICRoM.2015.7367772).
- [3] D. Zenteno-Lara, «Clase 9: Postura Relativa de un Cuerpo Rígido,» en *Robótica: de la Cinemática al Control*, UdeMy, México, 2020. dirección: <https://www.udemy.com/course/robotica-de-la-cinematica-al-control/>.
- [4] H.-L. Pham, V. Perdereau, B. V. Adorno y P. Fraitse, «Position and orientation control of robot manipulators using dual quaternion feedback,» en *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2010, págs. 658-663. DOI: [10.1109/IRoS.2010.5651097](https://doi.org/10.1109/IRoS.2010.5651097).
- [5] L. Sciavicco y B. Siciliano, *Modelling and Control of Robot Manipulators* (Advanced Textbooks in Control and Signal Processing). Springer London, 2012, ISBN: 9781447104490.
- [6] A. Rao, *Dynamics of Particles and Rigid Bodies: A Systematic Approach*. Cambridge University Press, 2006, ISBN: 9780521858113.
- [7] P. Matthews, *Vector Calculus* (Springer Undergraduate Mathematics Series). Springer London, 2000, ISBN: 9783540761808.
- [8] G. Trenkler, «The Moore-Penrose inverse and the vector product,» *International Journal of Mathematical Education in Science and Technology*, vol. 33, n.º 3, págs. 431-436, 2002. DOI: [10.1080/002073902760047922](https://doi.org/10.1080/002073902760047922). eprint: <https://doi.org/10.1080/002073902760047922>. dirección: <https://doi.org/10.1080/002073902760047922>.
- [9] A. Ben-Israel, T. Greville y C. M. Society, *Generalized Inverses: Theory and Applications* (CMS Books in Mathematics). Springer, 2003, ISBN: 9780387002934.
- [10] K. Lynch y F. Park, *Modern Robotics*. Cambridge University Press, 2017, ISBN: 9781107156302.
- [11] A. D. Luca, «Dynamic model of robots: Lagrangian approach,» en *Robotics II*, Dipartimento di Ingegneria Informatica Automatica e Gestionale Antonio Ruberti, Roma, Italia, 2022. dirección: https://www.diag.uniroma1.it/deluca/rob2_en/03_LagrangianDynamics_1.pdf.
- [12] S. Chapra y R. Canale, *Métodos numéricos para ingenieros*. MCGRAW-HILL, 1999, ISBN: 9789701020081.
- [13] L. Tsai, *Robot Analysis: The Mechanics of Serial and Parallel Manipulators* (A Wiley-Interscience publication). Wiley, 1999, ISBN: 9780471325932.
- [14] A. Codourey, «Dynamic Modeling of Parallel Robots for Computed-Torque Control Implementation,» *The International Journal of Robotics Research*, vol. 17, n.º 12, págs. 1325-1336, 1998. DOI: [10.1177/027836499801701205](https://doi.org/10.1177/027836499801701205). eprint: <https://doi.org/10.1177/027836499801701205>. dirección: <https://doi.org/10.1177/027836499801701205>.
- [15] E. G. Sheila Widnall John Deyst, «Lecture L26: 3D Rigid Body Dynamics: The Inertia Tensor,» en *Dynamics / Aeronautics and Astronautics — MIT Course No. 16.07*, MIT OpenCourseWare, Cambridge MA, 2008. dirección: <https://ocw.mit.edu/courses/16-07-dynamics-fall-2009/>.
- [16] D. Zenteno-Lara, «¿Qué es un Controlador PID? Un acercamiento a su principio matemático y funcionamiento,» en *Control de Sistemas Dinámicos*, UdeMy, México, 2019. dirección: <https://bit.ly/3tLtmEQ>.

- [17] D. Zenteno-Lara, «Estabilidad de Sistemas Dinámicos: De los Polos y Ceros a la Teoría de Estabilidad de Lyapunov,» en *Control de Sistemas Dinámicos*, UdeMy, México, 2019. dirección: <https://bit.ly/3td5NU1>.